NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

THREE AXIS FORCE OVERRIDE RATE CONTROL OF A PUMA 560 MANIPULATOR

by

Larry P. Ondrey

March, 1993

Thesis Advisor: Morris R. Driels

93-16962

93 7 28 062

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	Approved for public release; distribution is unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School	6b. OFFICE SYMBOL (If applicable) 34	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000		7b. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000

8a. NAME OF FUNDING/SPONSORING ORGANIZATION	8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)		10. SOURCE OF FUNDING NUMBERS

Program Element No.	Project No.	Task No.	Work Unit Accession Number

11. TITLE (Include Security Classification)
THREE AXIS FORCE OVERRIDE RATE CONTROL OF A PUMA 560 MANIPULATOR

12. PERSONAL AUTHOR(S) Larry P. Ondrey

13a. TYPE OF REPORT Master's Thesis	13b. TIME COVERED From To	14. DATE OF REPORT (year, month, day) March 1993	15. PAGE COUNT 126

16. SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES

FIELD	GROUP	SUBGROUP

18. SUBJECT TERMS (continue on reverse if necessary and identify by block number)
Force Control, Robot Control

19. ABSTRACT (continue on reverse if necessary and identify by block number)

This paper discusses the implementation of unilateral force override of rate control of a PUMA 560 robot manipulator in three degrees of freedom. A control system is developed utilizing the necessary sensors, hardware, and software interface to enable one to operate the manipulator in rate control with unilateral force control override. A review of the theory behind such a controller is conducted and stability issues addressed. a comparison of experimental results with the theoretical results is conducted and a simple program is developed to simulate the manipulator's response. These simulation results are compared to the experimental data.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS REPORT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassifed
22a. NAME OF RESPONSIBLE INDIVIDUAL Driels, Morris R.	22b. TELEPHONE (Include Area code) (408) 656-3384	22c. OFFICE SYMBOL ME/Dr

DD FORM 1473, 84 MAR   83 APR edition may be used until exhausted   SECURITY CLASSIFICATION OF THIS PAGE
All other editions are obsolete   UNCLASSIFIED

THREE AXIS FORCE OVERRIDE RATE CONTROL OF A PUMA 560 MANIPULATOR

by

Larry P. Ondrey
Lieutenant, United States Navy
B.S., United States Naval Academy, 1984

Submitted in partial fulfillment
of the requirements for the degree of

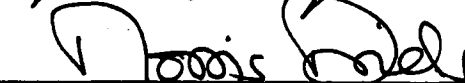## MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
March 1993

Author: _____
Larry P. Ondrey

Approved by: _____
Morris R. Driels, Thesis Advisor

_____
Matthew Kelleher, Chairman
Department of Mechanical Engineering

ii

ABSTRACT

This paper discusses the implementation of unilateral force control override of rate control of a PUMA 560 robot manipulator in three degrees of freedom. A control system is developed utilizing the necessary sensors, hardware, and software interface to enable one to operate the manipulator in rate control with unilateral force control override. A review of the theory behind such a controller is conducted and stability issues addressed. A comparison of experimental results with theoretical results is conducted and a simple program is developed to simulate the manipulator's response. These simulation results are compared to the experimental data.

DTIC QUALITY INSPECTED 5

Accesion For		
NTIS CRA&I		☒
DTIC TAB		☐
U announced		☐
Justification		
By		
Distibution /		
Availability Codes		
Dist	Avail and / or Special	
A-1		

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENT

I. INTRODUCTION

Force control of robot manipulators has received much attention during recent years. This is due in part to the growing interest in the use of robot systems to interact with undefined environments in carrying out tasks. The ability to measure and control the forces between a manipulator and the environment provides greater flexibility to the system when such interactions with the surroundings are required or desired. Traditional manipulator position control methods present limitations to a robot system in force control. If **rate** and **position control** of a robot are used exclusively, there is a need for thorough knowledge of the surroundings and extreme precision in the control of the manipulator's position. The ability for the robot to interact with the environment without causing damage to the manipulator or surrounding objects is dependent on the stiffness of the manipulator and the environment. If the stiffness of each of these is high, damage is likely to occur when the robot's end effector comes in contact with an obstruction as it continues to try and reach the desired position or maintain the desired rate.

The optimum characteristics of each type of controller differ considerably. Most robot systems in use utilize position control and are very stiff in order to provide

1

precision in positioning and speed in their motions. On the other hand, if one desires precise control of the forces a manipulator applies to an object, a manipulator of low stiffness would be advantageous. This allows a reasonable response speed to be maintained and lower contact forces to develop as a result of position errors. This conflicting requirement in stiffness for the two control methods presents a difficult design problem, where a compromise must be made.

Several applications can make use of this type of control approach. Almost any use of tele-operated robotics systems and autonomous vehicles can benefit from this control structure. The space shuttle RMS is currently operated using rate control which limits its capabilities. With the addition of a force override to the rate control system it could perform several tasks which currently require astronauts. The space shuttle RMS could be utilized to retrieve a satellite with the application of this control system rather than requiring several astronauts to perform a space walk. Other uses include assembly tasks of components in hazardous or sensitive environments in which human interaction is not directly possible or desired. This includes work on high voltage electrical systems, assembly of parts in a controlled atmosphere, and assembly or disassembly tasks in the nuclear industry.

The goal of this research is to extend the ability to control a manipulator using force override rate control to

three degrees of freedom (DOF) from a previously demonstrated single DOF.[Ref. 1] Control of forces in three cartesian space directions will be performed utilizing a PUMA 560 manipulator arm. The control system will be developed and analyzed for stability and performance. The system will also be simulated with a one DOF model and compared with theoretical results of a simple one DOF linear model.

II.   THEORY

A.   ROBOT CONTROL

A review of the basic manipulator control problem will be conducted before developing a force override rate controller for a PUMA 560 manipulator. This review will then be expanded to include a discussion of force control of manipulators in the next section and finally, the development of a force override rate controller in a third section.

1.   Control Methods

Controlling a robot manipulator can be categorized into three general areas. These include joint motion control, resolved motion control, and adaptive control. Each of these methods are currently utilized in control systems for existing manipulators and each has advantages and disadvantages. Joint motion control concerns the most elementary aspect of controlling a manipulator in that it deals with the control of each individual joint.   All control strategies must incorporate this aspect of robot control in some way. Resolved motion control involves developing control algorithms based on a cartesian or other useful coordinate system more easily related to by an operator in a given situation. Adaptive control utilizes some type of model of the manipulator as well as any environmental constraints to

control the manipulator. Each of these methods are discussed below.

2. Joint Space Control

Until recent years most applications of robot controls have used controllers that treated each joint of a multi-joint manipulator as a separate, independent system. Fu, Gonzalez, and Lee develop the equations of motion and transfer function for a linearized model of a single revolute joint. [Ref. 2] Figure 2.1 provides a relatively complete block diagram of a single electric motor driven joint. The block diagram notation is defined as follows:

J is the effective inertia of the mechanical joint and servo motor combined.

L_A is the servo motor inductance.

R_A is the servo motor resistance.
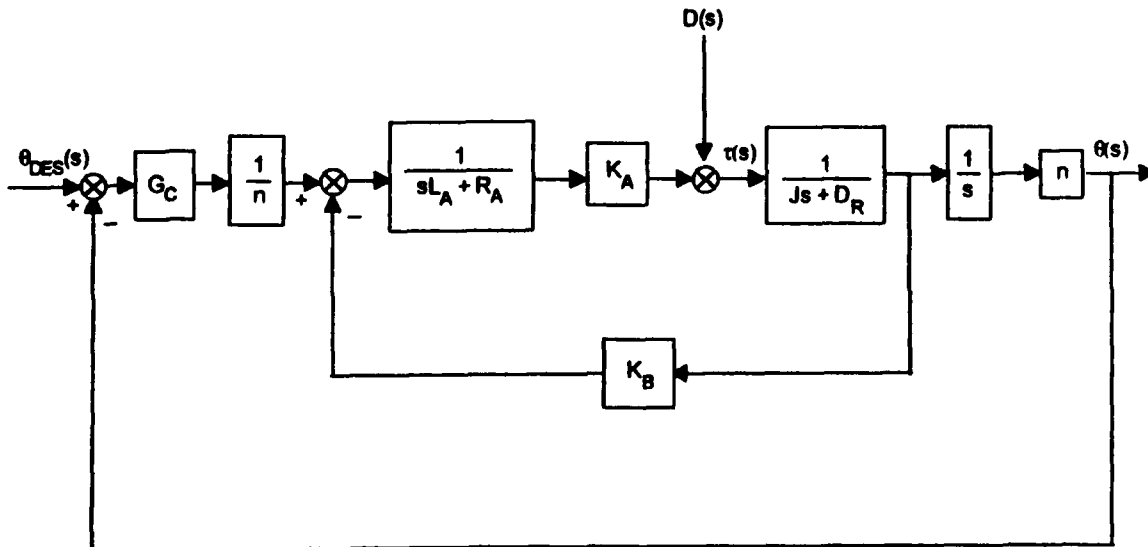
Figure 2.1. Single Joint Control Block Diagram.

D_R is the viscous damping in the joint.

n is the gear ratio between the servo motor rotation angle and the joint angle.

D(s) is any external disturbances applied to the system.

G_C is the position controller transfer function.

K_A is the proportionality constant between applied voltage and output torque for the servo motor.

K_B is the back electromotive force of the servo motor resulting from it's motion.

$\Theta_{DES}(s)$ is the desired joint angle.

$\Theta(s)$ is the actual joint angle.

$\tau(s)$ is the torque applied by the joint motor.

The following equation gives the closed loop transfer function for this system.

$$\frac{\Theta(s)}{\Theta_{DES}(s)} = \frac{G_C K_A}{JL_A s^3 + (JR_A + D_R L_A) s^2 + (R_A D + K_A K_B) s + G_C K_A} \quad (2.1)$$

G_C for a PID position controller can be represented by

$$G_C = K_P + K_V s + \frac{K_I}{s} \quad (2.2)$$

where K_P is a proportional gain constant, K_V is a derivative gain constant and K_I is an integral gain constant. Assuming the electrical time constant of the servo motor is small the term L_a can be neglected, leaving

$$\frac{\Theta(s)}{\Theta_{DES}(s)} = \frac{G_C K_A}{JR_A s^2 + (R_A D_R + K_A K_B) s + G_C K_A} \quad (2.3)$$

6

Substituting in the terms for G_c and performing some algebra results in

$$\frac{\Theta(s)}{\Theta_{DES}(s)} = \frac{(K_V K_A/JR_A)\, s + K_P K_A/JR_A + \dfrac{K_I K_A/JR_A}{s}}{s^2 + (R_A D_R + K_A K_B + K_V K_A)/JR_A)\, s + K_P K_A/JR_a + \dfrac{K_I K_A/JR_A}{s}} \quad (2.4)$$

This is a third order system with G_c containing all three control terms. If K_I is small, as is often the case in practice, the system closely follows the response of a second order system represented by

$$\frac{\Theta(s)}{\Theta_{DES}(s)} = \frac{(K_V K_A/JR_A)\, s + K_P K_A/JR_A}{s^2 + ((R_A D_R + K_A K_B + K_V K_A)/JR_A)\, s + K_P K_A/JR_A} \quad (2.5)$$

K_P and K_V can be selected to provide the desired response which is usually designed to be critically damped or slightly underdamped with a natural frequency considerably lower than the manipulator's structural natural frequency.

In a multi-joint robot D(s) includes gravitational loading as well as the reaction torques resulting from the motion and drive torques of the rest of the manipulator's joints. These disturbances are a complex function of the manipulator's position and motion. Additional disturbances would result from any constraints applied to the end effector by adding a payload or by an interaction with an environmental constraint, as is the case in a force control application.

Most current manipulator applications utilize constant values for K_p, K_v, and K_I in the joint servo controller. This is the case for the PUMA 560 manipulator. This control scheme results in a varying response of the manipulator with changes in manipulator's position, motion and loading.

Several algorithms have been developed that calculate the expected disturbances and compensates for them in the manipulator's control scheme. One such scheme, the computed torque technique, calculates the required torques needed at each joint, taking into account the interactions between the joints and with an added inertial load or other constraint at the end effector. This calculated torque is then utilized in the controller to drive the motions of the manipulator by using feedforward terms or some other control method. Craig develops the Newton-Euler and Lagranian formulations of the computed torque technique to establish the necessary drive torques of each joint for a desired motion.[Ref. 3] Although this type of control theoretically provides more uniform response, computational complexity makes this method difficult to achieve in practice. This control scheme also has the drawback of needing exact information concerning the manipulator's characteristics and environmental constraints in formulating drive torques for the joints. This is virtually impossible in many applications and the response still depends on the use of feedback control as formulated earlier.

An adaptive control scheme provides an optional control method without the drawbacks just discussed. This method is briefly described below.

3. Adaptive Control

The need for accurate models of the manipulator dynamics and environmental constraints often limit the ability to use control schemes like the computed torque technique. One solution to this problem is to use an adaptive control algorithm. Adaptive controls schemes compare the response of the actual manipulator to a reference model and the resulting difference between the two responses is in some way used to vary the feedback gains in the manipulator joint control loops. This essentially causes the manipulator dynamics to be modified until the manipulator dynamics match the reference model.

There are several methods developed to actually apply this method in practice and Fu, Gonzalez and Lee presents four of these.[Ref. 4] The ability to change the feedback gains of the joint servo controllers enables consistent response of the manipulator in a wide range of motions, payload conditions and constraints of the end effector motion. This type of control also requires considerable computational power in the robot's controller in order to perform the necessary algorithm that adjusts the feedback gains for the proper response. The PUMA 560's control algorithm does not

allow for this type of control and employing such a control strategy for the PUMA 560 requires essentially replacing the existing PUMA controller with a completely different system.

4. Resolved Motion Control

It is most convenient for an operator to control a manipulator in terms of a cartesian space reference frame. A reference frame with coordinates associated with the end effector, referred to as the tool frame, is most often utilized while a base or world frame, defined with respect to a stationary point in relation to the base of the robot, is sometimes used. The desired path defined in cartesian space must be related to the required motions of the manipulator's joints to achieve the motion. Craig [Ref. 5] briefly discusses the concepts of resolved motion control while Fu, Gonzalez, and Lee [Ref. 6] go into a more complete development of the control formulation. The control algorithm relies on the following equations which relate the coordinate kinematics of the cartesian reference frame to those of the manipulator joint motion.

$$\Theta_{DES}(t) = T^{-1}(X_{DES}(t)) \qquad \text{(2.6)}$$

$$\dot{\Theta}_{DES}(t) = N^{-1}(\Theta)\dot{X}_{DES}(t) \qquad \text{(2.7)}$$

$$\ddot{\Theta}_{DES}(t) = N^{-1}(\Theta)\dot{X}_{DES}(t) + N^{-1}(\Theta)\ddot{X}_{DES}(t) \qquad \text{(2.8)}$$

10

where:

N   is the Jacobian matrix for the manipulator.

T   is the coordinate transformation from joint space to cartesian space.

X_{DES} is the desired end effector position defined in cartesian space.

Θ_{DES} is the desired manipulator joint positions.

These equations indicate the computational complexity of this control method. Most existing manipulators controlled in cartesian space do not fully utilize these equations in their control scheme. The control system is often limited to performing the necessary computations to provide the inverse kinematics relating the joint positions to the cartesian space position, although resolved motion rate control and acceleration control which use the remaining two equations are possible. Fu, Gonzalez, and Lee also present the basic principles behind resolved motion force control which utilizes the following relationship to relate forces and moments to be applied by the manipulator's end effector, $F_{DES}(t)$, to the required joint torques, $\tau(t)$.[Ref. 7]

$$\tau(t) = N^T(\Theta) F(t) \qquad \textbf{(2.9)}$$

The PUMA 560 allows for two general motion control schemes. In the first method a motion command is related to the necessary joint position changes and each joint then moves as necessary to achieve the required new position in the

specified time allowed for the complete move. This is done independently of the other joints. This is referred to as joint-interpolated motion and is performed utilizing the MOVE command in the PUMA 560's VAL program language. The main disadvantage of this type of method is that it does not actually move the manipulator in a straight path from the initial position to the final position. The other control algorithm, referred to as straight-line motion results in the end effector moving in a straight-line path from the initial position to the final position. This is done by breaking the complete move into incremental motion segments. Each of these segments is converted to the necessary incremental positions each joint must attain. This scheme uses the MOVES command in the VAL program language and is used when the specific trajectory of the end effector is important. The drawback of this scheme is that greater computational complexity is involved which limits the speed of the manipulator's response. Further details of the motion control of the PUMA 560 will be considered in a later section.

B. FORCE CONTROL

1. Control Methods

Many force control strategies have been developed over the years. Whitney discusses several of these control strategies from an historical perspective.[Ref. 8]

Performance and system response differ for each method making each better suited for varying applications.

Force control strategies can be categorized into stiffness methods, damping methods , impedance methods, explicit force methods, and compliance methods. Each of these methods are discussed briefly below and evaluated as to applicability toward force override rate control of the PUMA manipulator.

a. Stiffness Control

Figure 2.2 shows a simple block diagram of a general model for stiffness control. The electrical time constant of the servo motor is neglected and the manipulator
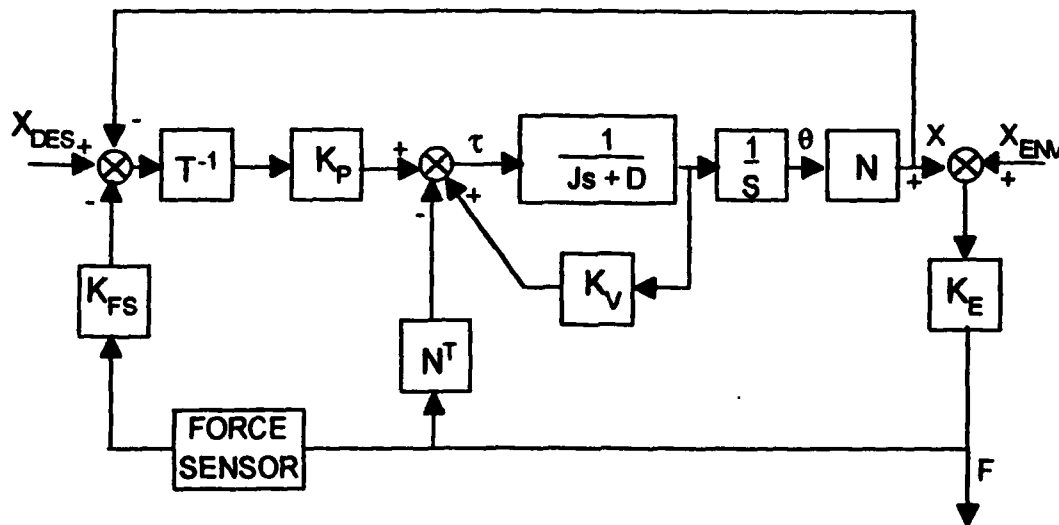
Figure 2.2. Stiffness Control Block Diagram.

is described by an inertial load, J, and a viscous damping term, D. K_{FS} is a force feedback stiffness gain, K_E represents the overall stiffness at the end effector/environment junction, X_{ENV} is the position of the environmental constraint, and all other terms are as defined previously.

Forces sensed at the end effector are multiplied by a gain matrix, K_{FS}, which results in a change in the desired position. The PUMA control architecture and programming language lends itself well to this type of control as the programming commands of the PUMA's VAL language are based primarily on position control of the manipulator joints to achieve the desired end effector position. To utilize this method for the PUMA 560, force sensors attached to the end effector provide the necessary feedback which is then used to update the position commands for the manipulator. This is essentially part of the method used in the force override rate controller to be described in detail in a later section.

b. Damping Control

Damping control is very similar to stiffness control but instead of utilizing force feedback to change the position command, the force feedback is used to change the commanded velocity of the manipulator. Figure 2.3 provides a block diagram of damping control with the additional term K_{FD} being a force feedback damping term. Forces sensed at the end effector are multiplied by K_{FD} which provides an input to the

14

commanded velocity. This method cannot be directly applied to the PUMA's control structure since the PUMA's programming structure relies primarily on position commands for control. However, this method can be adapted to the control structure to provide a form of rate control to the PUMA.
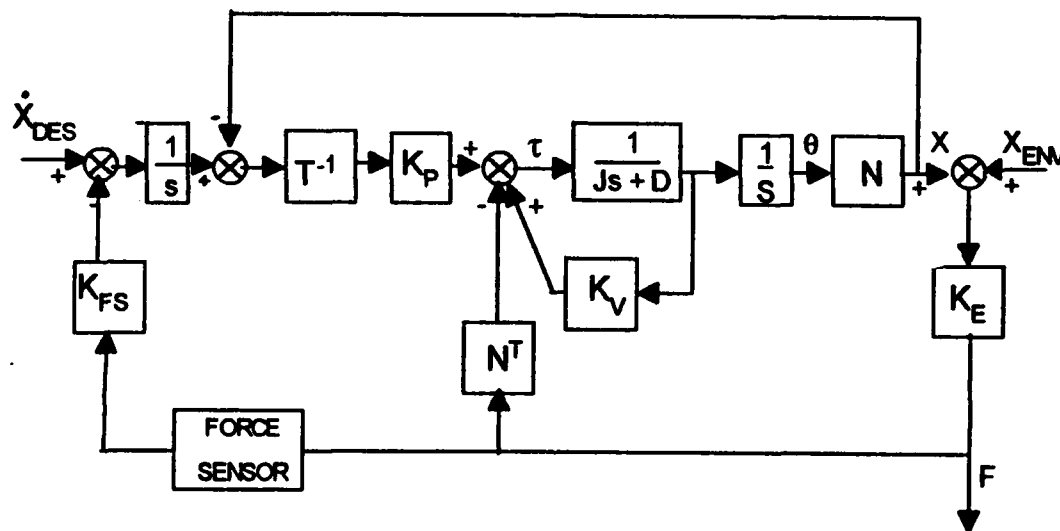
Figure 2.3. Damping Control Block Diagram.

Within the PUMA's programming capabilities is the ability to specify a speed at which a particular position order is carried out. This feature of the PUMA's control structure enables rate control of the manipulator to be performed in conjunction with position control. Details of this control structure are discussed in the next section.

c. Impedance Control

Impedance control is a combination of both stiffness control and damping control within one controller. Figure 2.4 provides a block diagram of such a controller.
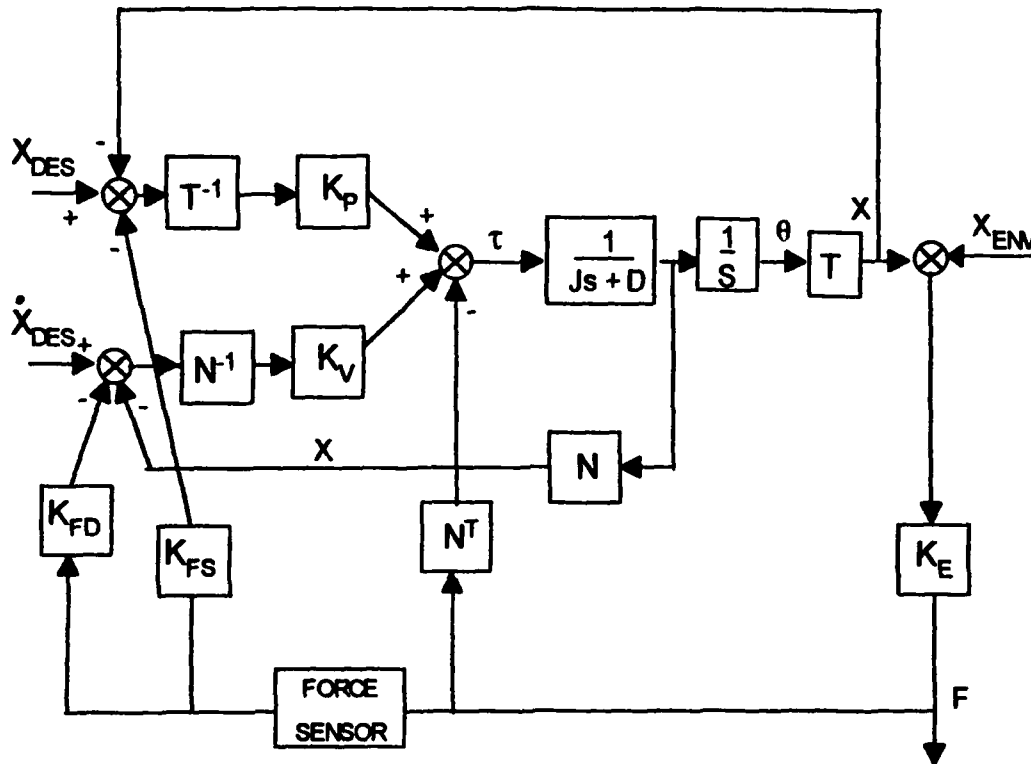
Figure 2.4. Impedance Control Block Diagram.

This combination results in PID force control which provides flexibility in controlling the manipulator's response. The force override rate controller to be developed in this paper is very similar to this form of control although it is not possible to implement this type of control exactly using the original PUMA control architecture and programming capabilities.

d. *Explicit force control*

Figure 2.5 shows a block diagram of an explicit force control method. In this method the desired forces are direct inputs to the control algorithm which are compared to
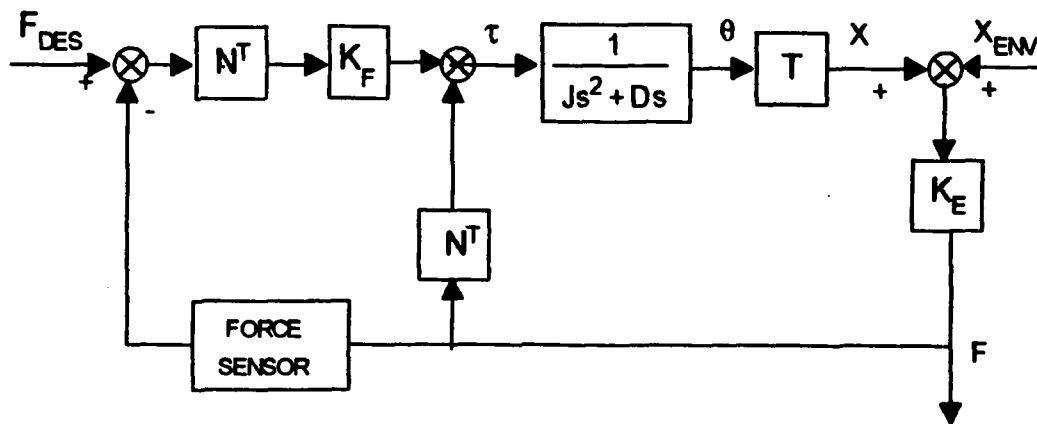
Figure 2.5. Explicit Force Control Block Diagram.

the forces fed back from a sensor located at the end effector. The force error e_F, is converted into joint space and multiplied by a gain, K_F, which produces the torque developed in each of the manipulator's joint servos. This is essentially a form of acceleration control. In the absence of any constraints, the manipulator will accelerate at a rate proportional to the force error between the desired force and the actual force developed at the end effector. This method cannot be directly implemented to control the PUMA as described, in which the joint servos are driven directly by

force errors. However, a control program could be written to indirectly apply this concept by converting the force error into an acceleration command, which is then integrated into rate and position commands that can be used to control the manipulator using the existing position control structure. When the manipulator is constrained, this method would provide a good force response with no steady state error. In unconstrained motion though, the position and rate of the robot is much more difficult to control. A force error in free space would give rise to an acceleration of the manipulator. To actually stop the manipulator at a desired position would be extremely difficult since it requires considerable operator input to attain the necessary desired force input to establish a zero commanded velocity. This does not even consider getting the end effector to stop at the desired location. Having said this, some experimental evaluation of this method is performed in an effort to validate these theoretical deductions.

e. Compliance Control

Figure 2.6 describes an example of compliance control. The algorithm is very similar to the stiffness control in that the force feedback is related to a change in the ordered position. In this case though, the rate of change of the forces developed at the end effector are also taken into account as shown by the term $K_s + K_D s$ in the feedback loop
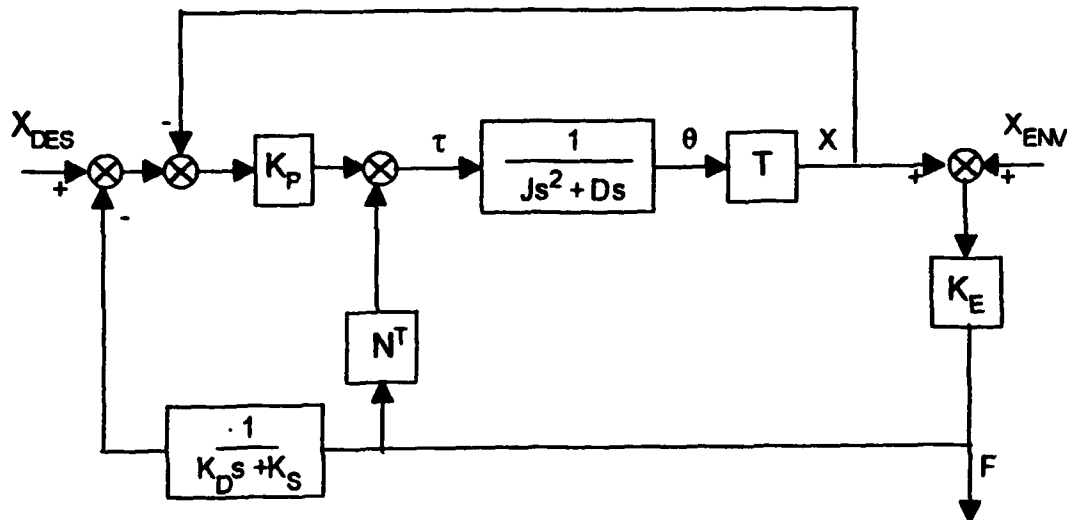
Figure 2.6. Compliance Control Block Diagram.

where K_S and K_D represent a desired manipulator compliance. Ishikawa, Sawada, Kawase, and Takata develop this type of controller for the PUMA 560 manipulator with variable position error gain, K_P, in the joint servo controllers. [Ref. 9] This variable gain requires essentially replacing the original computer hardware and software of the PUMA with three computer processors that perform all the computational tasks, and interface with the manipulator's joint servos to make the control system possible. This alteration provides the computational speed necessary to prevent unacceptable delays in implementing the control algorithm. The added feature of variable gain in the position controller enables stability to be maintained for virtually any constraint condition and desired manipulator compliance

19

parameters. While this provides more flexibility in the constraints the manipulator can handle, it still has the problem of compromising responsiveness for the sake of retaining stability, which is common to most force control strategies.

C. UNILATERAL FORCE OVERRIDE OF RATE CONTROL

1. Description

Figure 2.7 provides a block diagram for a simple model of a proposed force override rate controller for the PUMA 560 in one DOF. The previous definitions for the variables
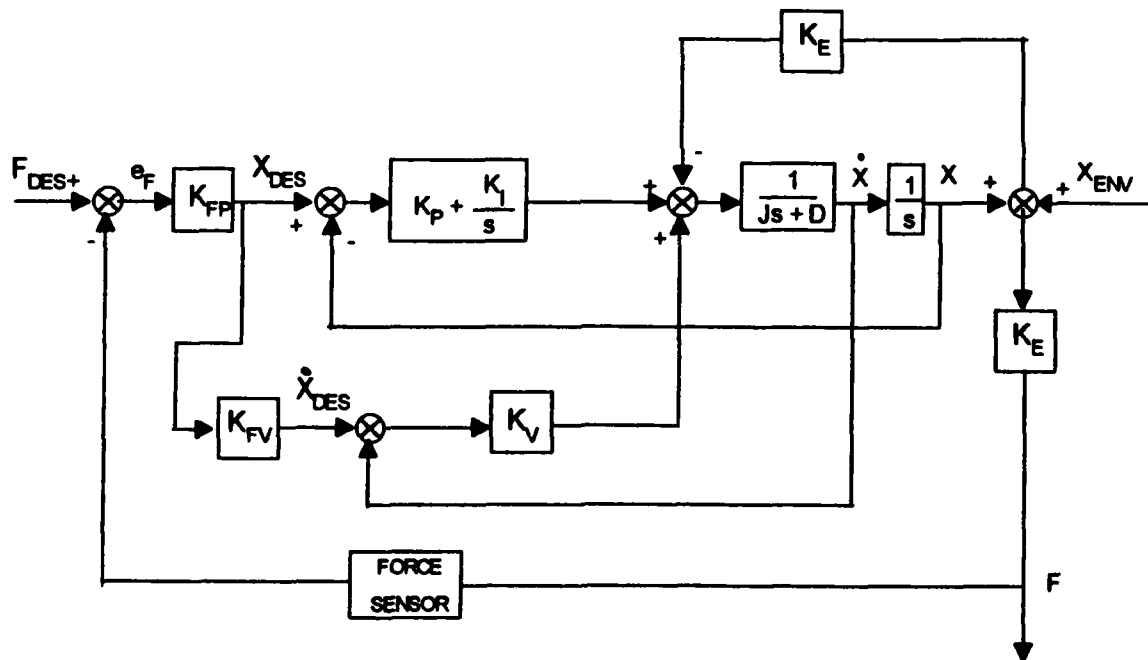
Figure 2.7. Force Override/Rate Control Block Diagram.

continues to apply with K_{FP} and K_{FV} representing force error position and velocity gains, respectively. The controller

20

consists of essentially two control loops. The inner loop contains the existing control system of the PUMA position controller and the outer loop provides position/rate control commands to the inner loop based on force errors. Figure 2.8 provides a description of the joystick and end effector used in this control system. Forces sensed at the joystick and the end effector are compared to provide a force error. This force error is then multiplied by position and rate gains to provide position and rate signals to the manipulator's controller. The PUMA 560's VAL programming language does not allow one to order a rate command without any position command, but does allow a rate command in conjunction with a particular position command. If no rate control command is provided with a position command the manipulator is programmed to carry out the position command at a speed specified in the within the control architecture. This is one reason for $\dot{X}_{DES}$ being dependent on X_{DES} as shown in the block diagram. This linear model of the block diagram does not model the complete control structure of the PUMA 560 but provides some insight into the expected behavior. A simulation is developed in a later section which more accurately emulates the control structure of the PUMA 560, taking into account time delays in the control program.
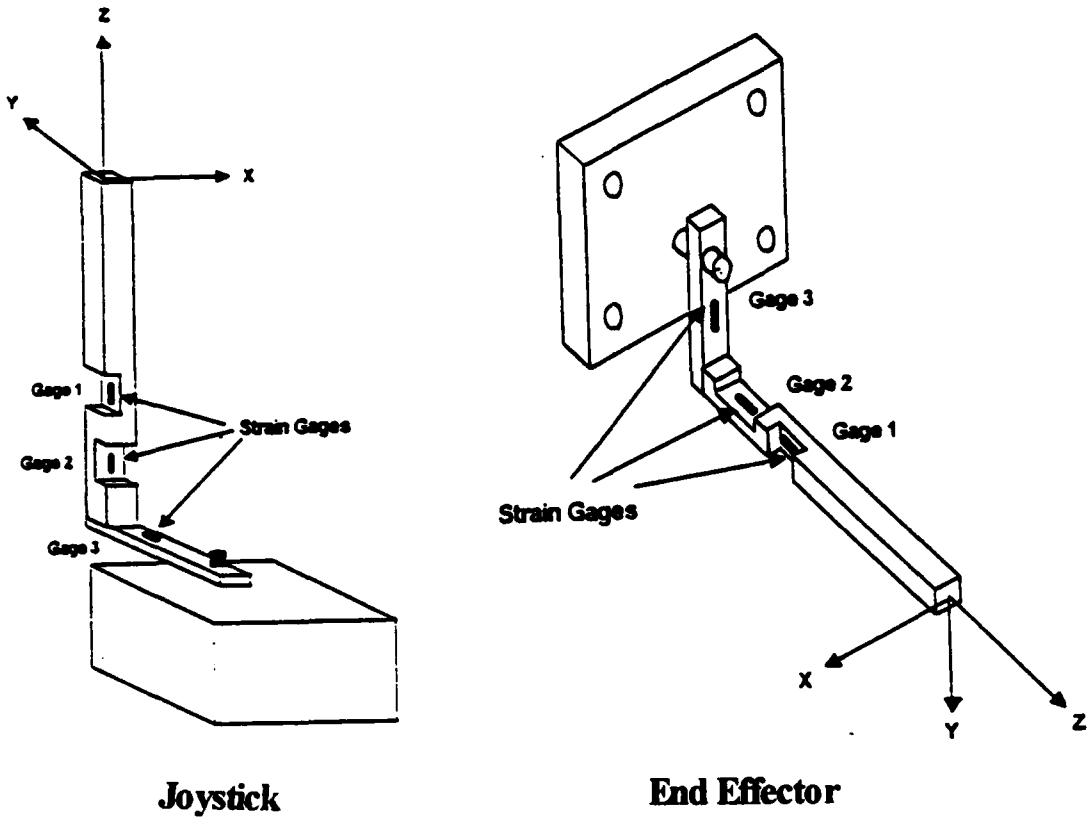
Figure 2.8. Joystick and End Effector.

2. Analysis

Simplifying the system by assuming a value of zero for K_{FV} results in the following closed loop transfer function with respect to forces:

$$\frac{F(s)}{F_{DES}(s)} = \frac{K_{FP}K_P K_E + K_{FP}K_E \dfrac{K_I}{s}}{Js^2 + Ds + K_E + K_P + K_{FP}K_P K_E + K_{FP}K_E \dfrac{K_I}{s}} \quad (2.10)$$

Defining the following variables relating the environmental stiffness to the position feedback gain and the force error gain to the environmental stiffness.

$$K_{FP} = \frac{\beta}{K_E} \qquad (2.11)$$

$$K_E = \alpha K_P \qquad (2.12)$$

Making the necessary substitutions the transfer function becomes

$$\frac{F(s)}{F_{DES}(s)} = \frac{\beta(K_P s + K_I)}{Js^3 + Ds^2 + (\alpha + \beta + 1)K_P s + K_I(\beta + 1)} \qquad (2.13)$$

K_I is assumed to be small the system has a second order transfer function of

$$\frac{F(s)}{F_{DES}(s)} = \frac{\beta \omega_n^2}{s^2 + 2\zeta \omega_n s + (\alpha + \beta + 1)\omega_n^2} \qquad (2.14)$$

with the following definitions

$$\omega_n = \sqrt{\frac{K_P}{J}} \qquad (2.15)$$

$$2\zeta \omega_n = \frac{D}{J} \qquad (2.16)$$

Ishikawa, Sawada, Kawase, and Takata experimentally determined values for the characteristics of the PUMA 560 in their development of a compliance controller and these values are as follows: [Ref. 10]

K_p = 85000 N/m

ω = 125.7 rad/s

ζ = 1.0 (Assumes critical damping)

Figure 2.9 provides a Bode plot of the system using values of α = 0.015 and β = 1.0. This value of α closely approximates the experimental values for K_p and K_E while the value of β is arbitrarily set in the control program and can be changed. With simple proportional control of the force error, a steady state error in the actual applied force results from a simple step input.
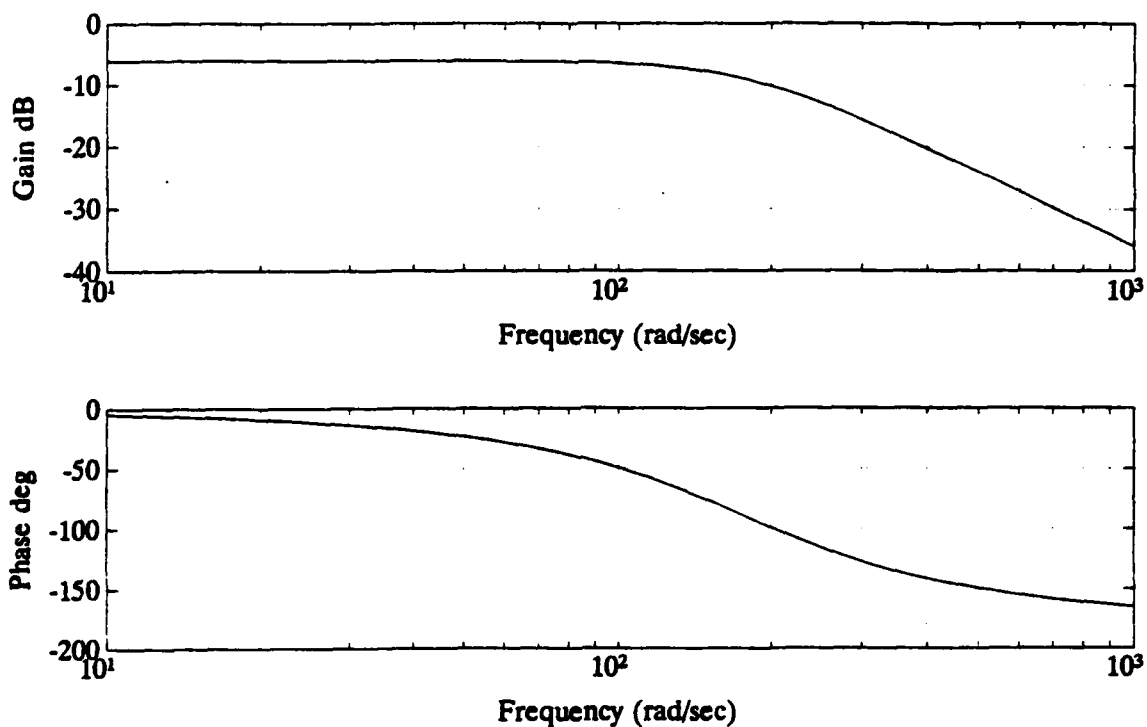
Figure 2.9. Bode Plot of Force Override/Rate Control.

From the transfer function this error to a step input can be expressed by

$$E_{ss} = \frac{\alpha + 1}{\alpha + \beta + 1} \qquad (2.17)$$

24

The steady state error and speed of response are directly related to the factors α and β, with β having the greater impact. A low value for α, corresponding to low overall stiffness of the system, is desired and a large value for β, corresponding to a large value of K_{FP}, results in a better response. The value of β is expected to help predict the stability of the system when computational delays are introduced to the system.

Replacing K_{FP} with $K_{FP}+K_{FI}/s$ results in proportional plus integral force control. This leads to a transfer function of

$$\frac{F(s)}{F_{DES}(s)} = \frac{\beta K_P s + \alpha K_{FI} K_P^2}{Js^3 + Ds^2 + (\alpha+\beta+1) K_P s + \alpha K_{FI} K_P^2} \qquad (2.18)$$

This eliminates the steady state error in force for a step input but results in difficult control of the manipulator in free space. Giving K_{FP} a value of zero results in a control structure essentially equivalent to the explicit force control method described previously. A force error will produce an acceleration of the manipulator rather than the desired rate control. This results in difficulty in controlling the manipulator's motion in free space. If contact between the manipulator and a constraint could be detected then a control algorithm could change the value of K_{FI} from zero in free space while in rate control, to some positive value when in contact

25

with a constraint to enable reducing the steady state force error. In the current application only force error signals are used as inputs to the control algorithm and the operator's visual observation is relied upon to determine when contact is made.

This analysis has been concerned with a single DOF model of the six DOF robot system. This enables a simple analysis of the system but results in inaccuracies. Eppinger and Seering perform an analysis of the effects of these manipulator model simplifications on the response and stability of force control.[Ref. 11] Not taking into account the complete system dynamics results in an inherently stable system model which when fully modelled can show unstable behavior.

III. PRELIMINARY WORK

A. SINGLE DOF HYDRAULIC SYSTEM

The idea of controlling a manipulator such as the Space Shuttle RMS in rate control with force override stems from a desire to be able to use rate control to properly position the manipulator with respect to an object and then automatically transition to force control once contact is made. This would be particularly useful if one is working with a moving object such as would be the case with the Space Shuttle RMS interacting with a satellite. This concept of force override of the normal rate control of a manipulator has been tested using a single DOF hydraulic system [Ref. 12].

1. Description

Figure 3.1 provides a general description of the hydraulic force control system.[Ref. 13] An electro-hydraulic servo valve controls the fluid flow to a cylinder whose piston rod, representing an end effector, provides a linear displacement proportional to the fluid flow rate. A force sensor, consisting of strain gages mounted on a cantilever beam assembly, is attached to the end effector and measures forces developed between the end effector and an obstruction. The joystick consists of a control arm attached to a three way toggle switch. Strain gages are also mounted

27

on the joystick to detect forces applied to the joystick. Figure 3.2 provides a description of the joystick and end effector force sensors.
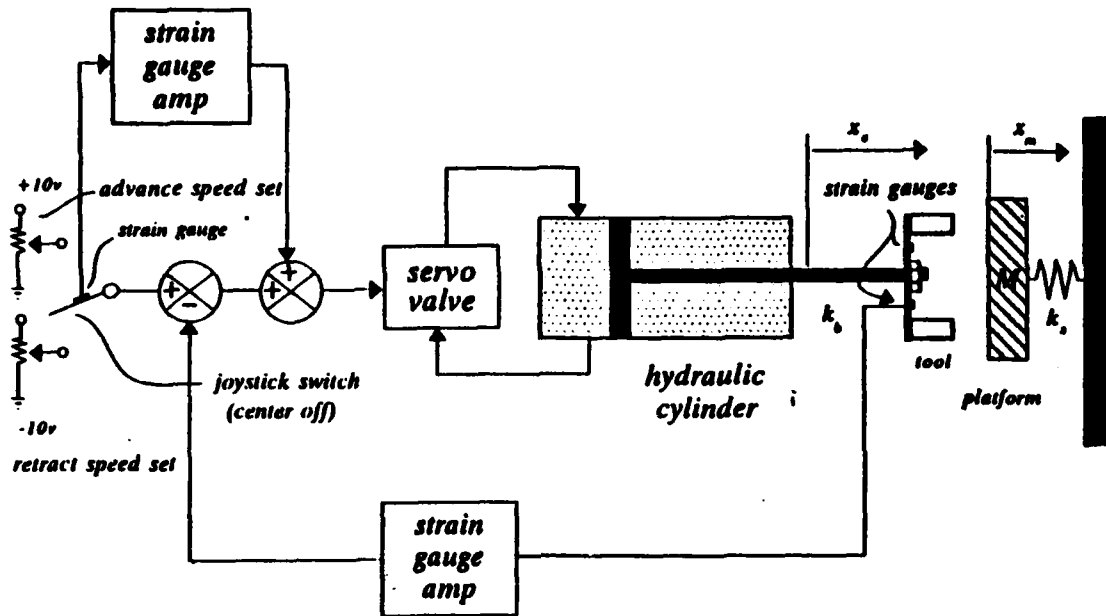
Figure 3.1.   Hydraulic Force Override Rate Control System. [Ref. 13]

Movement of the joystick controls the position of the toggle switch which controls a +10/-10 V voltage supply to the servo valve.   In the mid-position no voltage is supplied to the servo valve.   Moving the joystick in a forward direction supplies a positive voltage and moving it in a reverse direction supplies a negative voltage.   The hydraulic system is arranged so that a positive voltage supply to the servo valve results in forward motion of the end effector and a negative voltage supply results in reverse motion of the end
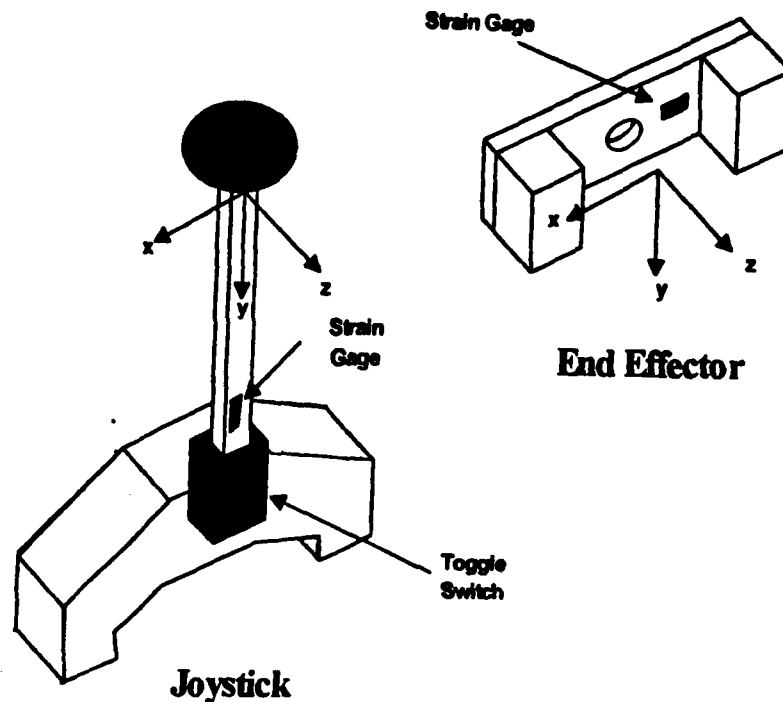
Figure 3.2. Single DOF Joystick and End Effector.

effector. With no forces acting on the joystick and no forces
acting against the end effector the piston moves at a constant
rate in a direction based on the voltage supplied to the servo
valve.

Voltages from the strain gage amplifiers of the force
sensors on the joystick and the end effector are added to the
original +10/-10 V supplied by the positioning of the joystick
to provide force override control of the initial rate control.
When the end effector comes into contact with an obstruction,
a voltage opposing the control voltage is developed by the end
effector force sensor. This is summed with the existing

controlling voltage to the servo valve and reduces the control voltage. This reduces the speed of the end effector's motion. This continues until the control voltage to the servo valve returns to a zero value and motion stops. An additional force applied to the joystick will develop a voltage in the force sensor attached to the joystick. When added to the existing control voltage supplied to the servo valve, this additional voltage will cause a non-zero voltage to the servo valve and continued motion of the end effector. This motion continues until the feedback voltage from the force sensor on the end effector counteracts the voltage due to the applied force on the joystick.

2. Analysis and Testing

The hydraulic system acts as a valve controlled piston whose dynamics are developed in [Ref. 14]. Figure 3.3 provides a simple block diagram of the system described above. The following notation is used:

E_d :   The initial voltage threshold supplied to the servo valve when the joystick is positioned in the forward or reverse direction. This may be adjusted by the operator and essentially establishes the initial rate of piston motion with no environmental constraint.

F_a :   The applied force on the joystick.

F :   The actual force being applied by the end effector on the environment.

E_f :   The sum of the threshold voltage and voltage due to force applied on the joystick.
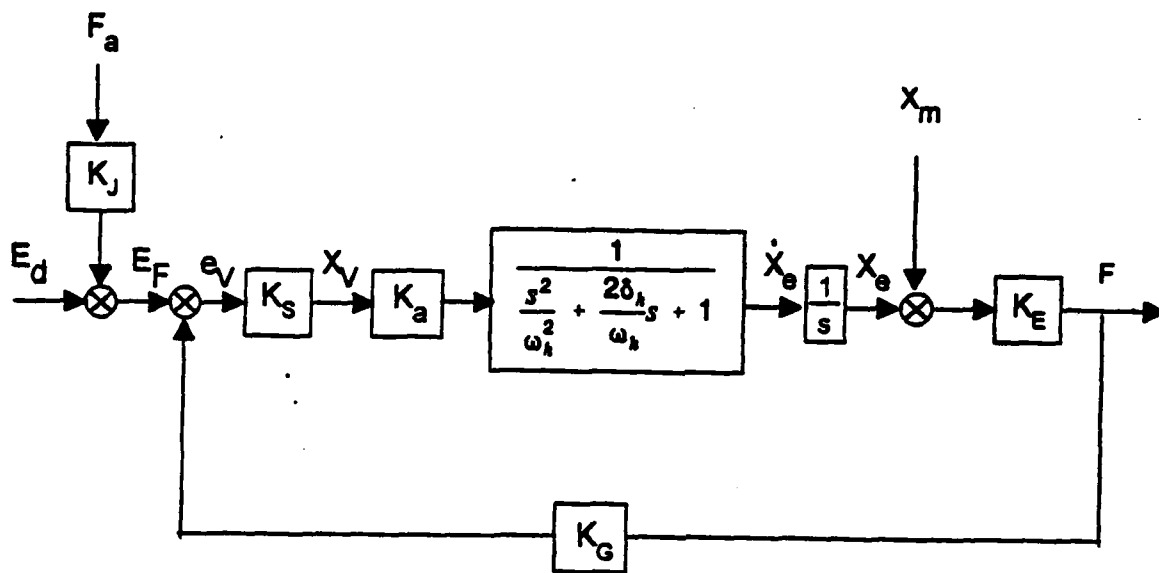
Figure 3.3. Block Diagram of Hydraulic Control System.

e_v : The voltage error developed from force sensors and the threshold voltage which is applied to the servo valve.

x_e : The position of the end effector with respect to an some stationary reference frame.

x_m : The position of the mass-spring providing an environmental constraint with the respect to the stationary reference frame.

x_v : The position of the servo valve relative to the null position.

K_a : A proportionality constant relating the servo valve position to the force applied on the hydraulic piston by the hydraulic fluid.

ω_h : The natural frequency of the hydraulic piston and load arrangement.

δ_h : The damping ratio of the hydraulic piston and load arrangement.

K_E :   The overall stiffness of the end effector and spring-mass assembly.

K_s :   A proportionality constant indicating the relationship between the voltage applied to the servo valve and the position of the servo valve.

K_G :   The strain gage amplifier gain of the end effector force sensor.

K_J :   The strain gage amplifier gain of the joystick force sensor.

The diagram assumes that the dynamics of the servo valve are sufficiently fast to be negligible and the servo valve's position is simply proportional to the input voltage. Also, the disturbance to the piston dynamics due to the reaction force at the end effector is neglected. The closed loop transfer function of the system is:

$$\frac{F(s)}{E_F(s)} = \frac{K_s K_a K_E \omega_h^2}{s^3 + 2\delta_h \omega_h s^2 + s + K_G K_s K_a K_E \omega_h^2} \qquad (3.1)$$

This is a third order system in which the value of K_G can be selected to establish the desired sensitivity of the end effector to force interactions. The value of K_J is selected to match the value of K_G so the added force applied at the end effector matches the added force applied at the joystick. This system is essentially the same as the single DOF system model developed in Chapter II. The dynamics of the system are modelled slightly differently and the initial threshold voltage is not present in the theoretical model. Qualitative

32

tests confirm that the system response matches the expected response of this analysis.

B. SINGLE DOF FORCE CONTROL OF A PUMA MANIPULATOR

1. Description

The control structure of the hydraulic system is applied to the PUMA 560 to provide force override of rate control in a single direction. Figure 3.4 provides a general diagram of the system.[Ref. 15] The joystick assembly and force sensor on the end effector from the hydraulic system are used in controlling the PUMA 560. For this system however, an analog to digital (A/D) conversion circuit is required to enable data from the force sensors of the joystick and end effector to be communicated to the PUMA 560's control computer. The A/D converter is used to convert the error signal developed between the joystick and the end effector, not the actual signals developed by each device. This same conversion circuit is also used in the three DOF force override rate controller and is described in more detail in the next chapter.

In this system the PUMA 560 and it's control computer replaces the hydraulic cylinder and electro-hydraulic servo valve. The direction of control is along the z axis of the tool frame of the manipulator.
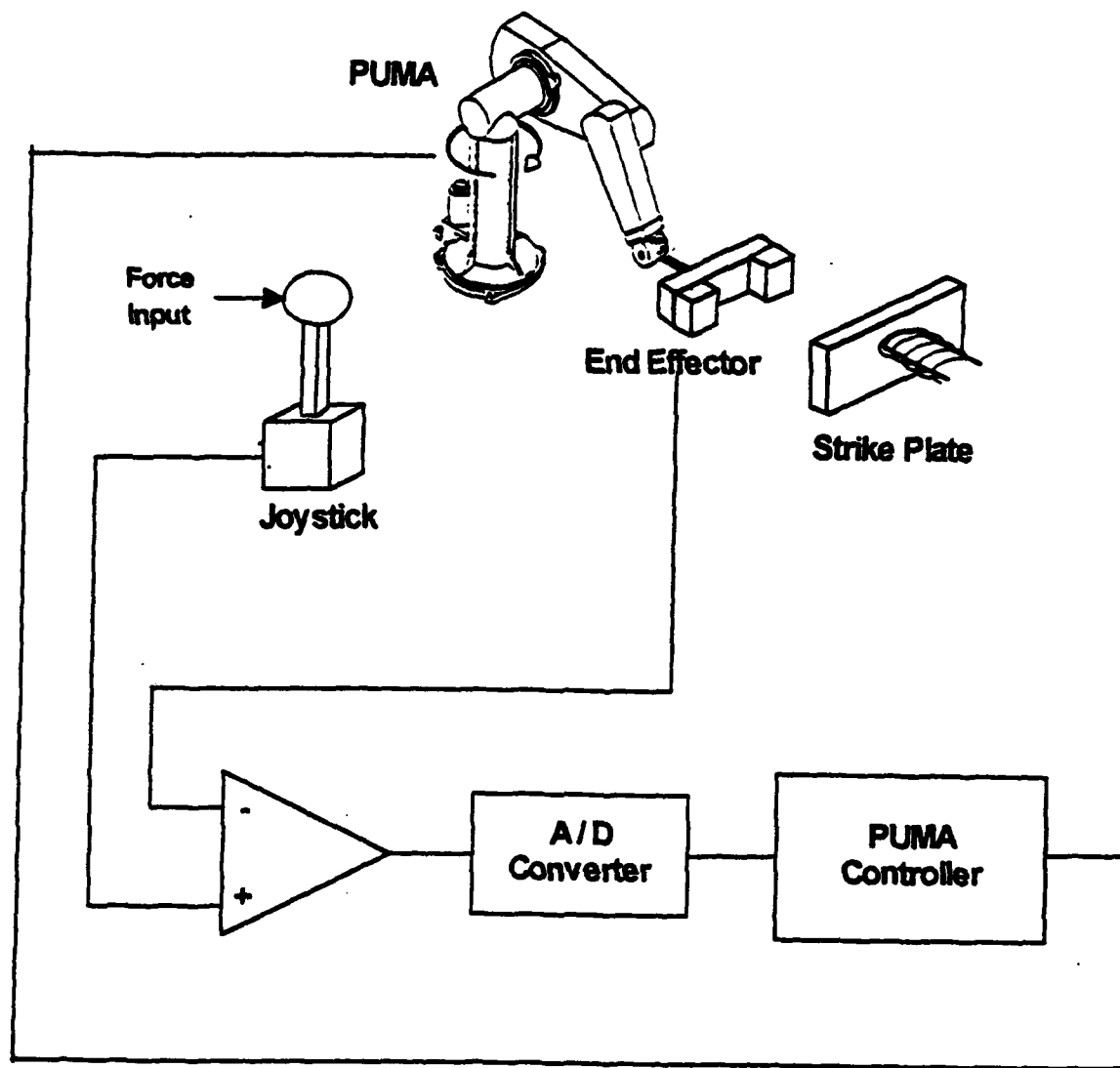
Figure 3.4. One DOF Force Override Rate Control of a PUMA 560 Manipulator.[Ref. 15]

2. Operation and Testing

Implementing this control method for the PUMA requires developing the necessary software to obtain information from the force sensing circuit and to control the motion of the manipulator based on this input. Driels develops the control

algorithm for this system using the PUMA 560's VAL programming language.[Ref. 16] A demonstration of the system is performed which verifies the feasibility of the control system. Qualitative results indicate that the system performs as expected.

This leads to the development and testing of a control system for the PUMA 560 manipulator that provides force override rate control in three DOF. The remaining chapters are devoted to this endeavor.

IV. THREE DOF FORCE OVERRIDE RATE CONTROLLER FOR A PUMA 560

A. DESCRIPTION

1. Overall System

The control system for force override rate control of the PUMA 560 manipulator consists of the original PUMA 560 control system with the force control system built around it. Figure 4.1 provides an overall description of the system. The PUMA 560's control system consists essentially of a PID controller designed to control the position and velocity of the tool tip.

The force control system added to the PUMA control system consists of an end effector and joystick with strain gages mounted on each to sense forces in each direction, the associated strain gage amplifiers, a summing and biasing circuit, an A/D converter, and a control program written in VAL. Each item is discussed in some detail below.

2. End Effector and Joystick

Rather than purchase sophisticated and expensive commercially available force sensors for the system, relatively simple sensors consisting of strain gages mounted on an L-shaped beam are fabricated to test the feasibility of the control system. Figure 4.2 provides a drawing of the end effector and joystick. Each is made of aluminum and the
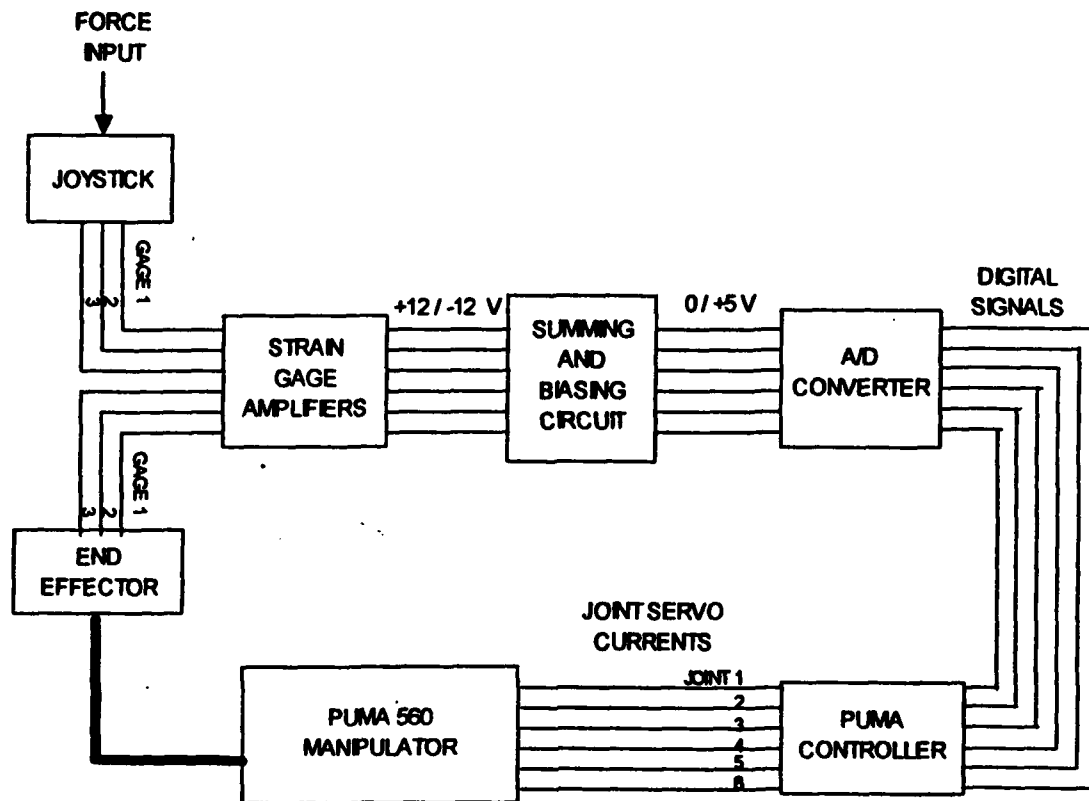
Figure 4.1. Three DOF Force Override Rate Controller.

dimensions are chosen so as to provide relatively low stiffness for good stability of the force control system.

Strain gages are mounted as shown in Figure 4.2 to provide the ability to determine the forces applied by the end effector tip in each of the tool frame cartesian directions.

A calibration is performed to establish the desired voltage output from the strain gage amplifiers for an applied force. The calibration is conducted by placing or hanging weights on the end effector and joystick and adjusting the amplifier gain to establish the desired output. Weights of .25 lb (1.1 N), 0.5 lb (2.2 N), and 1.0 lb (4.4 N) nominally

End Effector
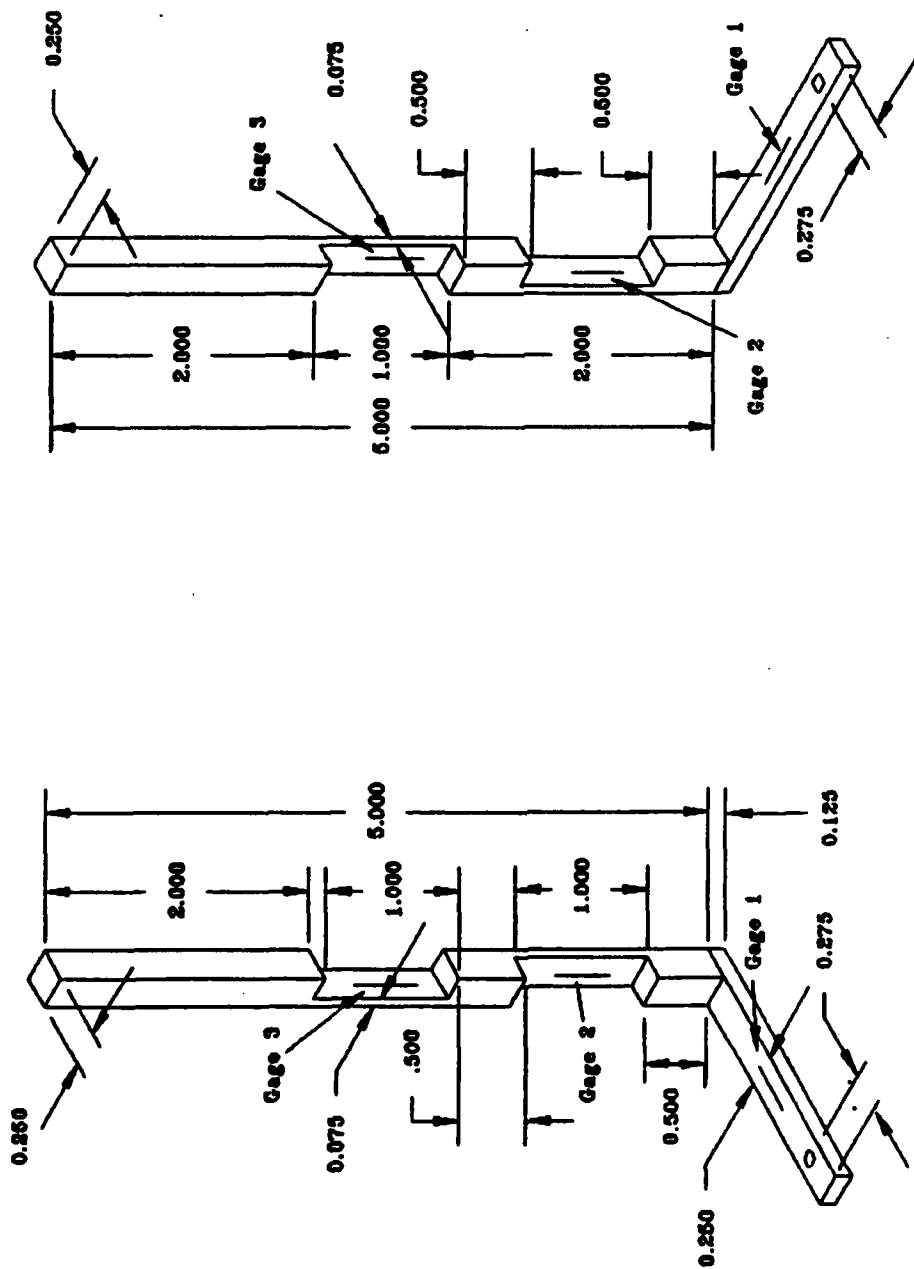
Joystick

Figure 4.2.   Joystick and End Effector.

are used for the calibration. Tables 4.1 and 4.2 provide the calibration results.

It is important to match the outputs of the end effector and joystick during the calibration since error signals between the two sensors are used in the control algorithm, not the actual signals from each sensor. This arrangement prevents compensation for calibration errors in the software but also reduces the amount of digital computation and data exchange required.

Strain gage amplifiers not specifically designed for this system are used due to their immediate availability. The amplifiers have a limited gain adjustment and are generally more sensitive than desired for this application. However, their output is of very high quality, having very little noise. The sensitivity of the amplifiers and the relatively low mechanical strength of the end effector and the joystick limit the magnitude of forces that can be developed and analyzed.

From Tables 4.1 and 4.2 one can see that there is considerable coupling between strain gage 3 and strain gage 2 for forces applied in the y direction. This creates problems in utilizing these sensors in the control algorithm. The amount of coupling that occurs is not the same for the end effector as it is for the joystick. A nominal force of 2 N applied in the y direction results in a voltage of 6.7 V in gage 3 of the joystick and only 6.0 V in gage 3 on the end

TABLE 4.1. JOYSTICK CALIBRATION DATA

Direction of Force	Strain Gage No.	Output Voltage (Volts)		
		Applied Load		
		4.4 N	2.2 N	1.1 N
+ x	1	8.0	4.1	2.0
	2	0	0	0
	3	0.40	0.23	0.12
+ y	1	-0.10	-0.05	-0.02
	2	12.0	6.0	2.94
	3	< -13.0	-6.7	-3.4
- z	1	0.40	0.13	0.08
	2	0	0	0
	3	-3.0	-1.5	-0.80

effector. These voltage outputs assume that gage 3 for both, the joystick and end effector, provides a 3 V output for a 2 N force applied in the z direction. This disparity cannot be resolved but this does not severely effect the testing of the system.

The stiffness of the joystick and end effector are estimated using solid mechanics principles of beam deflection and superposition. Approximations are made to limit the complexity of the analysis. Transverse loading effects of forces applied on the sensors are taken into account and pure bending is assumed. Torsional loading of lower beam, however, is taken into account when evaluating stiffness in the x

40

TABLE 4.2. END EFFECTOR CALIBRATION DATA

Direction of Force	Strain Gage No.	Output Voltage (Volts)		
		Applied Load		
		4.4 N	2.2 N	1.1 N
+ x	1	-8.0	-4.2	-2.0
	2	0	0	0
	3			
+ y	1			
	2	-12.0	-6.0	-2.95
	3	> +12.0	+6.2	+3.0
- z	1			
	2			
	3	+3.0	-+.5	+0.80

direction. Results of the analysis for stiffness are listed
in Table 4.3. The results indicate that the stiffness in the
y direction is approximately one half the stiffness in the x
direction and the stiffness in the z direction is very high
compared to the stiffness in the x and y directions. This
leads to control and stability problems which are discussed in
the next chapter.

The strain gage amplifiers utilize a two gage bridge
and develop an output in the range of -12/+12 V. The
amplified signal is sent to a summing and biasing circuit as
well as a data acquisition board used for recording force
data.

41

TABLE 4.3. END EFFECTOR AND JOYSTICK ESTIMATED STIFFNESS

Direction	Stiffness (N/mm)
x	1.205
y	0.525
z	104.3

3. Summing and Biasing Circuit

Figure 4.3 is a diagram of the summing and biasing circuit. This circuit produces a voltage signal based on the difference in the forces developed at the joystick and those developed at the end effector. The output of each strain gage on the joystick is summed with the output of the corresponding strain gage on the end effector. The strain gages of the end effector are connected in such a way that output voltages of its gages are opposite in sign from those of the joystick for a given applied force. This results in a voltage signal in the summing circuit proportional to the force error between the joystick and the end effector in the range of +12/-12 V. This signal is then converted to a range of 0 to +5 V in an analog biasing and attenuation circuit. This step is required for proper operation of the A/D converter which converts an analog voltage signal in this range.

4. A/D Converter

The A/D converter is a National Semiconductor ADC816 converter which allows for 16 channels of analog data
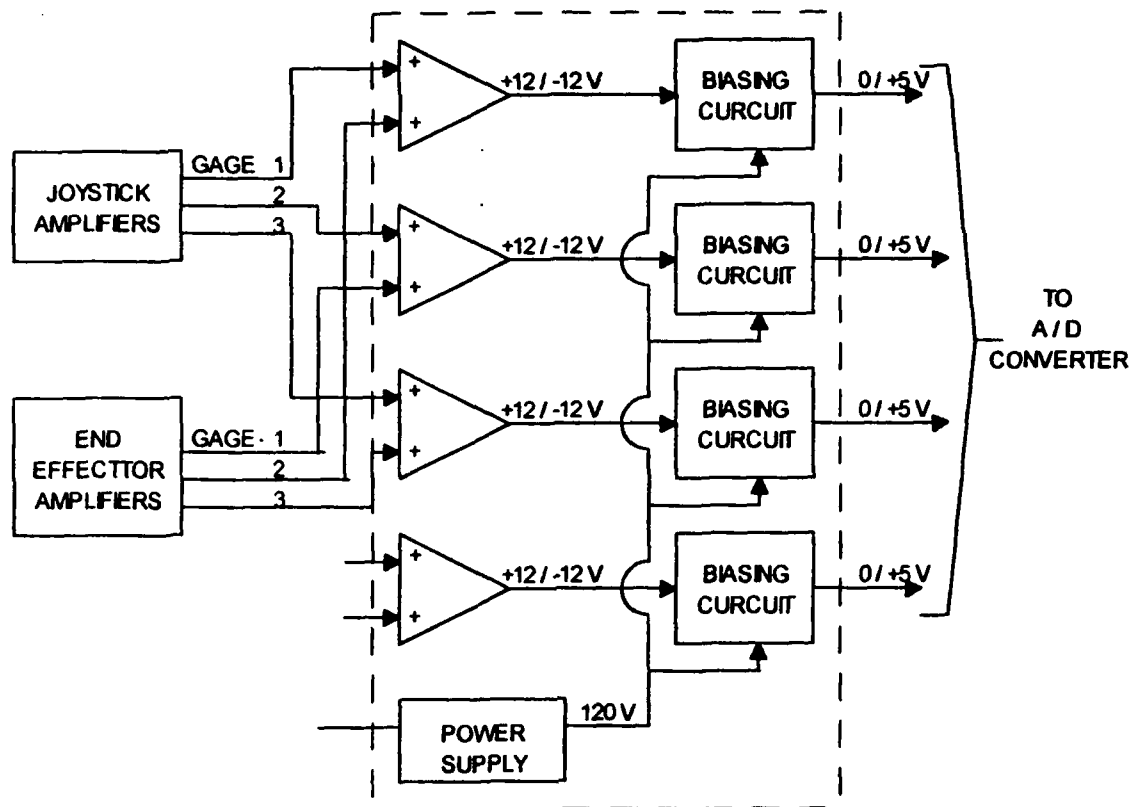
Figure 4.3. Summing and Biasing Circuit.

conversion to digital signals.[Ref. 17] The device uses power supplied by the PUMA controller and converts a 0 to +5V analog voltage input into an eight bit digital value. The converter is operated in free-run mode with an internal oscillator allowing 200 conversions/sec.[Ref. 18] The converter utilizes a multiplexer to select the channel to converted. Of the 16 available channels only four are wired for this application and only three of these channels are actually used. The output is wired to one of the four eight bit digital input/output (I/O) ports of the PUMA 560. controller. A diagram of the A/D converter is shown in Figure

43

4.4.[Ref. 19]   Figure 4.5 provides a wiring diagram
of the data conversion circuit cable interface with the PUMA
560 I/O port.[Ref. 20]

5. Computer Input/Output Interface

The PUMA 560 has the capability of transferring 32
bits of digital data using four I/O ports of one byte each.
The bits are numbered 1-32, with 1-8 assigned to I/O port 1,
9-16 assigned to I/O port 2, 17-24 assigned to port 3, and 25-
32 to port 4. The **SIGNAL** command in the VAL language is used
for output purposes while the **BITS** command is used for input.
A sample statement using the **SIGNAL** command is as follows:

SIGNAL 1,-2,3,-4

The numbers correspond to the bit number defined above and the
sign indicates the voltage state of that bit. A minus(-) sign
turns the bit off (a high voltage) while the plus(+) sign or
no sign turns the bit on (low voltage). This command controls
which channel of data the A/D converter uses as output to the
PUMA control program.

The input from the A/D converter is obtained using the
BITS command as in the following example:

BITS(1003,8)

The first number corresponds to the address of the first data
bit to be received, where 1001 corresponds to data bit 1, 1002
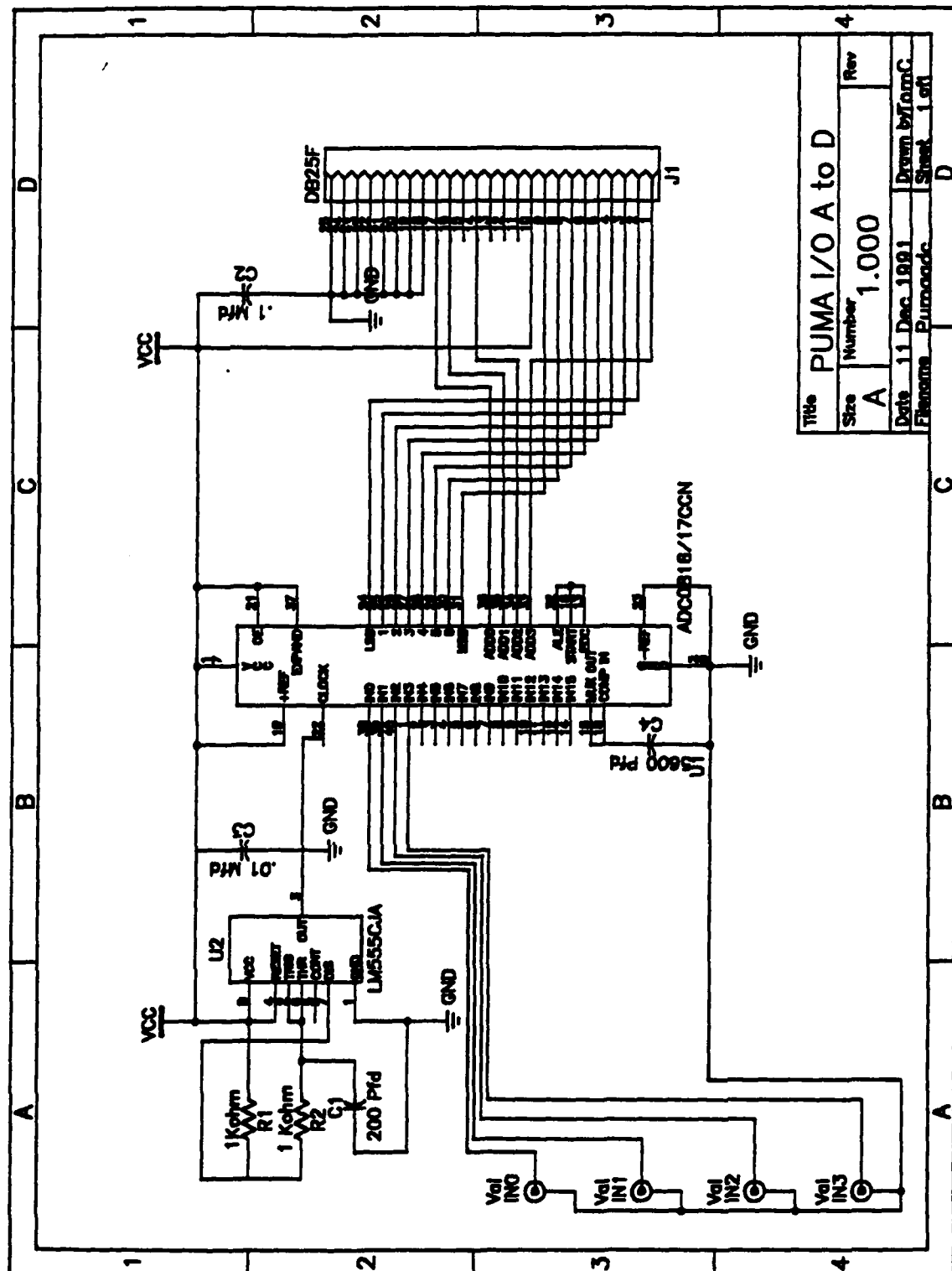
44

Figure 4.4. A/D Converter Diagram.[Ref. 19]

to data bit 2, and up to 1032 corresponding to bit 32. The second number indicates the number of data bits to be received. In this example statement the values of data bits 3 through 11 would be received.

PUMA
50 pin Centronics
Male

A / D Converter
25 pin DB Male

Input byte

D_0 13 → 2
D_1 12 → 3
D_2 11 → 4
D_3 10 → 5
D_4 9 → 6
D_5 8 → 7
D_6 7 → 8
D_7 6 → 9
26-50 → Any one of 18-25

Output byte

D_0 24 → 17
D_1 23 → 16
D_2 22 → 14
D_3 21 → 1
20
19
18
17 → 10

V_{cc}
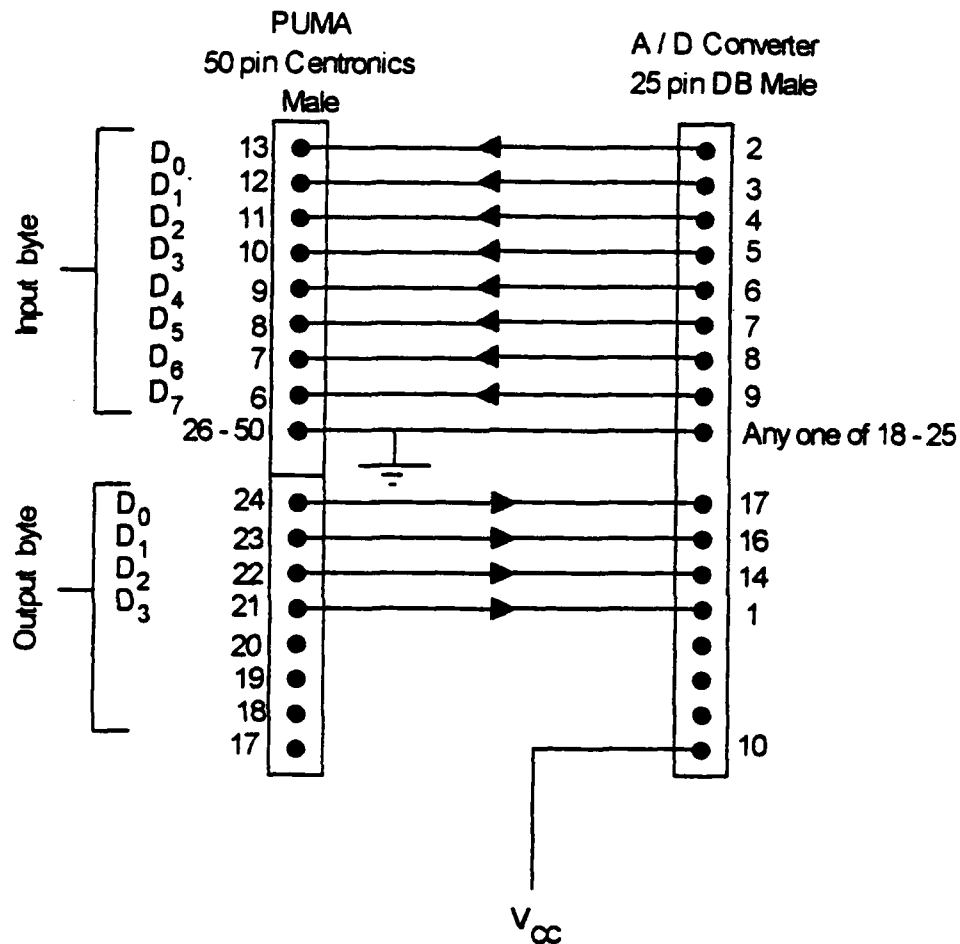
Figure 4.5.   A/D Converter Interface Cable Wiring Diagram.
[Ref. 20]

The Unimation Corporation does market an analog I/O board for use with the PUMA 560 but the equipment described

above was on hand and is utilize to save on expenses and provides more flexibility.

6. Environment

Four types of environmental constraints are used to evaluate the performance of the system. The same spring mass assembly used in the preliminary work of the hydraulic control system and in demonstrating the single DOF system of the PUMA manipulator are used to simulate a compliant stationary constraint. The aluminum block support for the spring mass assembly is placed in the manipulator's work space to provide a rigid stationary constraint. The joystick used in the preliminary work is also placed in the work space to provide a task oriented constraint with which to experiment with. Finally, a small mass placed on a low friction surface in the work space allows for testing of the manipulator in handling a simple inertial load.

7. Force Data Acquisition and Recording

The outputs of the strain gage amplifiers are wired into a +10/-10 V A/D data acquisition board which is connected to an IBM XT computer. This data acquisition board supports up to 16 channels of analog data. Six channels are utilized to record the outputs from the strain gage amplifiers into a data file. The processing speed of the computer allows for a data acquisition rate of approximately 85 Hertz for recording data. This is fast enough to obtain accurate force data from

the system since the system dynamics occur at a considerably lower rate. An effort is made to use the data acquisition system to perform some type of force display but limitations in the computer's processing speed, the monitor's capabilities, and the communication time required between these two devices prevent this.

8. PUMA 560 Control System

The PUMA 560 control system has a main computer processor which supplies motion commands to individual microprocessors which control each joint servo motor. Figure 4.6 provides a diagram of the system structure.

Each joint is controlled as a basic servomechanism with a Rockwell 6503 microprocessor which uses a PID control law to position the joint.[Ref. 21] The integral control can be deactivated with the **INTOFF** VAL command. Each joint follows the dynamics developed for joint space control in Chapter II. This inner loop operates with a 0.875 msec sample time and each joint controller receives an updated position command from the main controller every 28 msec.

The main computer is a DEC LSI 11/02 processor. [Ref. 22] It performs the primary functions of processing user motion commands entered in the VAL programming language and coordinating the actions of the six joint servo controllers. The main computer performs all coordinate transformations between joint space and cartesian space
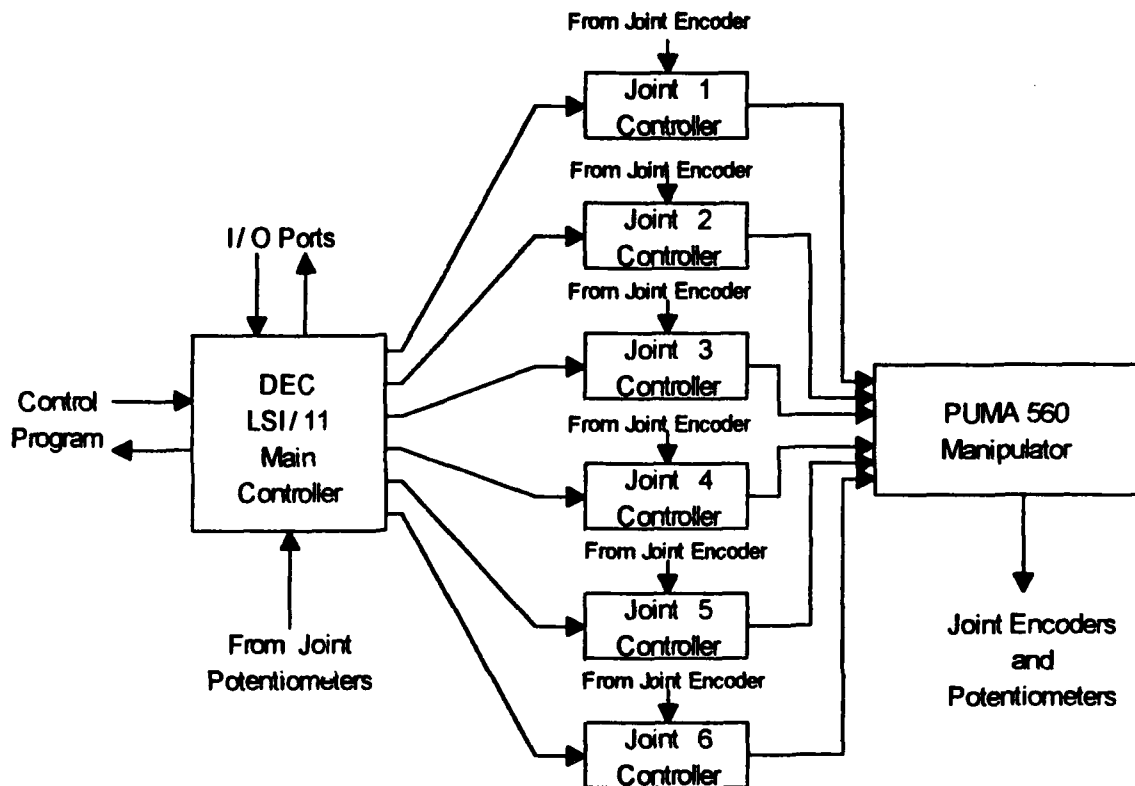
Figure 4.6. PUMA 560 Control System Diagram.

positions and develops incremental position commands to the individual joint servos every 28 msec.[Ref. 23] These position commands are based on motion commands entered through user programs. The processor looks ahead to the next motion command in order to coordinate continuous path motion. [Ref. 24] The motion of the manipulator remains continuous by adjusting from one commanded position to the next. Considerable computational time is required for this and if a particular motion does not take enough time the manipulator will not be able to develop a continuous path trajectory for the next motion command. This results in jerky

motion as the manipulator stops momentarily between individual positions. The continuous path trajectory planning can be deactivated to stop the robot after completing each motion command.

The computer also has the capability to perform straight line motion or joint interpolated motion as discussed in chapter two. Considerably more computations are required to handle the transformations involved with straight line motion. Each commanded motion must take approximately 140 msec in order to maintain continuous path motion while about 60 msec is needed while using joint interpolated motion. [Ref. 25] For this reason joint interpolated motion is used in this application where small position changes are used and the difference between a straight line trajectory and joint interpolated trajectory is minimal.

9. Control Program

Appendix A contains a copy of the VAL program written to implement the three DOF force override rate control algorithm with the PUMA 560 manipulator. It consists of three general sections. The initial section has the user enter values for the various control gains and provides some option as to the structure of the control system. The next section initialize the force error values from force error circuits. The final section is a continuous loop which provides position

and rate commands to the main controller. Figure 4.7 provides a flow diagram of the control program.

The operator initializes the system control parameters which includes specifying the value of ß, as defined in Chapter II, to be used for each direction and whether or not integral control is to be used within the joint servo control loops. The **NONULL ALWAYS** command of step 22 deactivates the normal processing sequence in which the main processor waits for reports from each servo processor that the current incremental position has been reached before a new position is ordered. This reduces delays in the program execution to allow a faster response and less time lag in processing motion commands.

All of the adjustable control system gains are also defined in this first section. The value of **adfactor** is a conversion of the values received at the PUMA's I/O port to the voltage outputs from the summing circuit. The values of **xamp.gain**, **yamp.gain**, and **zamp.gain** provide conversions from the voltages out of the summing circuit to the actual force errors between the end effector and joystick. Steps 32 through 34 set the values of **xforce.gain**, **yforce.gain**, and **zforce.gain** which correlate to the term K_{FP} in the block diagram of Figure 2.7. Step 44 specifies a value for **movetime**, an estimate of the required of each motion command to maintain continuous path motion. Step 45 defines **maxspeed** as an estimate of the maximum linear velocity the PUMA 560 can

51
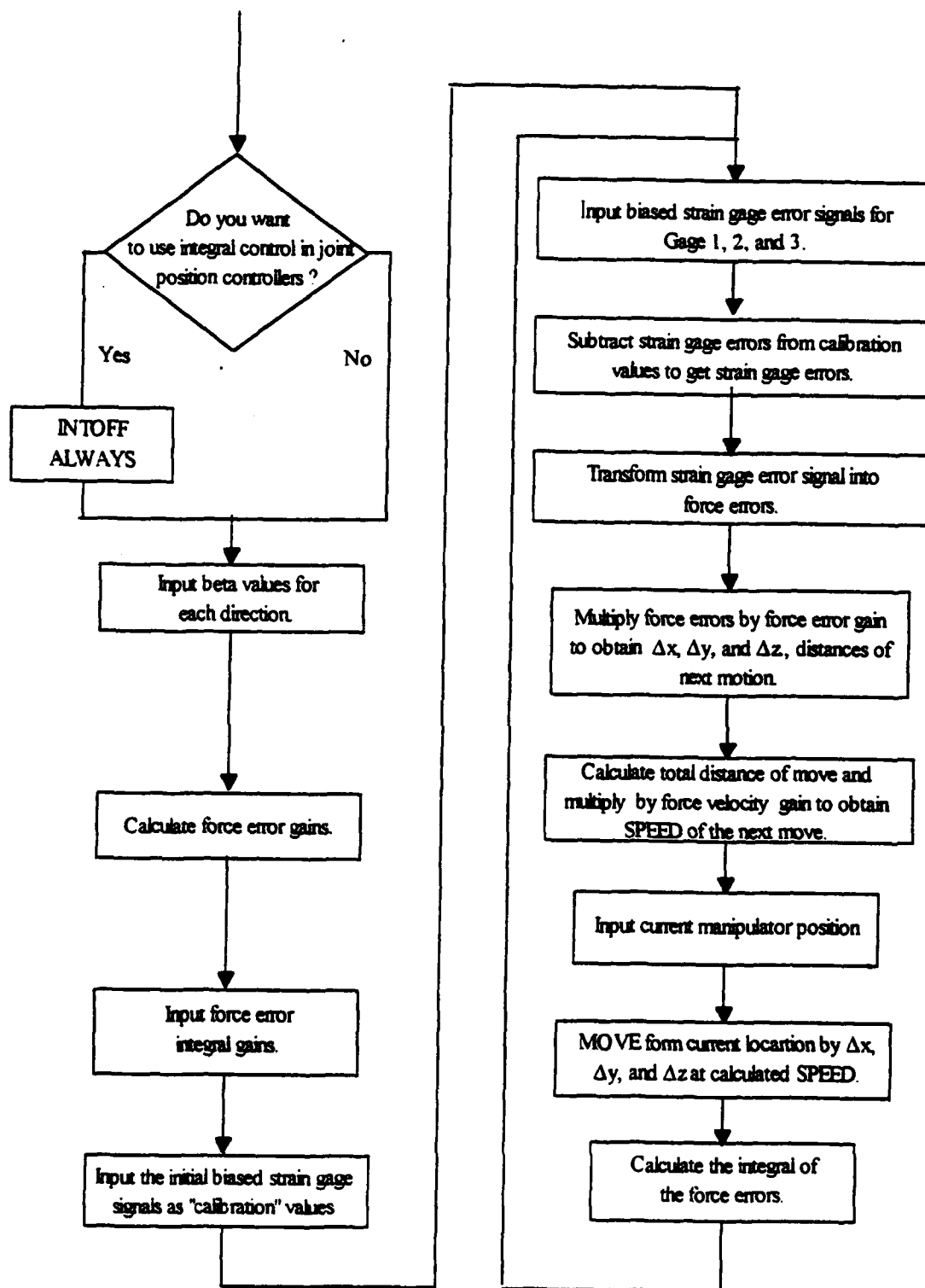
Figure 4.7. Control Program Flow Diagram.

move in mm/sec. This value is based on tabulated PUMA operating parameters ([Ref. 26]) and is needed in step 98 to provide a basis for specifying a speed for the next motion.

The second section goes through a series of steps which obtain initial values from the A/D converter corresponding to force error signals. This section begins with step 52 and continues through step 68. These values are assumed to be the force error signals when no forces are acting on the end effector or the joystick. If the system is adjusted perfectly with the zero adjustment on the strain gage amplifiers set just right the values of **xcal, ycal** and **zcal** would be 128. This section provides a means to effectively zero the force errors without having to physically change the zero adjustments on the strain gage amplifiers. This does not prevent one from needing to manually calibrate the gain adjustments on the strain gage circuits as discussed before. The sequence of steps is explained below.

Step 52 is a **SIGNAL** command which sends a signal from the PUMA controller through I/O port 1 to the A/D converter specifying the A/D converter to convert channel 0 to a digital value for output to the PUMA's I/O port. A short **FOR - NEXT** loop provides a delay before a **BITS** input command is used to assign the value of the current channel selected for output in the A/D converter to **xcal**. The delay is experimentally determined to be necessary to allow the A/D converter's

multiplexer enough time to switch to the desired channel before its output is obtained using the **BITS** command. This sequence of steps is repeated twice to obtain the values of **ycal** and **zcal** from channel 1 and 2 of the A/D converter, respectively.

The final section starts at line 72 which initiates the force error input sequence. The values for **xin, yin,** and **zin** are assigned values using the same sequence of steps used to obtain the values of **xcal, ycal,** and **zcal** which are the current values of the force error signals at the start of each control loop. In steps 87 through 92 the **xin, yin,** and **zin** are subtracted from the initial calibration values of **xcal, ycal,** and **zcal,** respectively and the values are decoupled to obtain force errors in the three cartesian directions defined by the manipulators tool frame. In steps 93 through 95 the force errors are multiplied by a force gain and the integral of the force error is multiplied by a corresponding gain (if integral control of forces is used) to obtain the distance the manipulator is to move in each direction. These values are assigned to **xmove, ymove,** and **zmove.** The overall distance is calculated in step 96 and a speed is calculated in step 97 so that the motion will take the required time to allow continuous path motion. An actual linear speed cannot be specified with joint interpolated motion but the speed must be defined relative to the *normal* speed defined internally within the VAL controller. This

relative speed is calculated in step 98 which provides the commanded speed for the next motion command. The *normal* speed of each manipulator joint is defined in the PUMA manual.[Ref. 27] Step 99 defines a coordinate transformation based on the distance the manipulator needs to move in each direction and assigns it to **delta**. Step 100 assigns a coordinate transformation defined by the current manipulator position to **initial** and step 101 is a joint interpolated motion command for the manipulator to move to the location defined by adding the **delta** transformation to the **initial** position. Steps 102 through 104 numerically integrate the force error in each direction using a simple Euler method and assign the values to **inxferr, intyferr**, and **intzferr**.

B. EXPERIMENTAL TESTS

1. Force Control of Stationary Constraint

a. Description

Several tests are performed testing the response of the system when interacting with the compliant and rigid stationary constraints described earlier. Tests are performed to evaluate the manipulator's response in each of the three cartesian directions. The manipulator is aligned a nominal distance of 5 mm away from the constraint and a step force input is applied to the joystick. This force input is applied by hanging the same weights used to calibrate the joystick on the end of the joystick. This provides a relatively steady

force input without having to substitute a test signal for the joystick strain gage inputs. The joystick is oriented in such a way that the gravitational loading of the weight is applied in the desired direction. The constraint is also positioned to ensure that the constraint is perpendicular to the direction of the commanded force and manipulator motion. When testing the response in the z direction the compliant spring-mass system is utilized to ensure adequate compliance in an otherwise extremely stiff environment and the manipulator is positioned at a point just prior to contact with the constraint.

System response is obtained for step force inputs of different magnitudes and with different values of β used in the control program. The response is obtained for tests in which integral control of the individual joint position servo controllers is activated and when it is de-activated. Tests are also run to determine the response of the system when integral control of the force error signal is used. During these tests the manipulator is positioned so as to just be making contact with the stationary constraint.

Tests are also run to check the system response to a relatively low frequency sinusoidal force input. The strain gage amplifier output from the joystick in the x direction is replaced by a sinusoidal voltage from a function generator. The test is run at two frequencies and varying values of β to demonstrate the variations in magnitude and phase lag of the

force applied by the end effector to the rigid constraint. The coupling between the outputs of the strain gages in the y and z directions make it difficult to perform this test in either of these directions and tests in these directions are not performed.

b. Results

(1) Step Input

Figure 4.8 shows the force response to a 0.5 lb (2.2 N) nominal force command in the x direction with $\beta=0.5$ and using PID control in the individual joint servos. The top graph is the response in the x direction, the middle graph is the response in the y direction, and the bottom graph is the response in the z direction. Figures 4.9, 4.10, and 4.11 show the response for $\beta=1$, $\beta=2$, and $\beta=4$ with PID control used in the individual joint servo position controllers. Figures 4.12, 4.13, and 4.14 show the results for a force input in the y direction while using the same system control parameters. Figure 4.15 shows the response of the system to a force input of 4.0 N in the z direction with $\beta=20$, $\beta=40$, and $\beta=60$ in the z direction, respectively, and $\beta=1$ in the x and y directions. The top graph is of $\beta=20$, the middle graph is for $\beta=40$, and the bottom graph is for $\beta=60$. Using values of β much lower than these in the z direction results in a very slow response due to the high end effector stiffness in this direction.
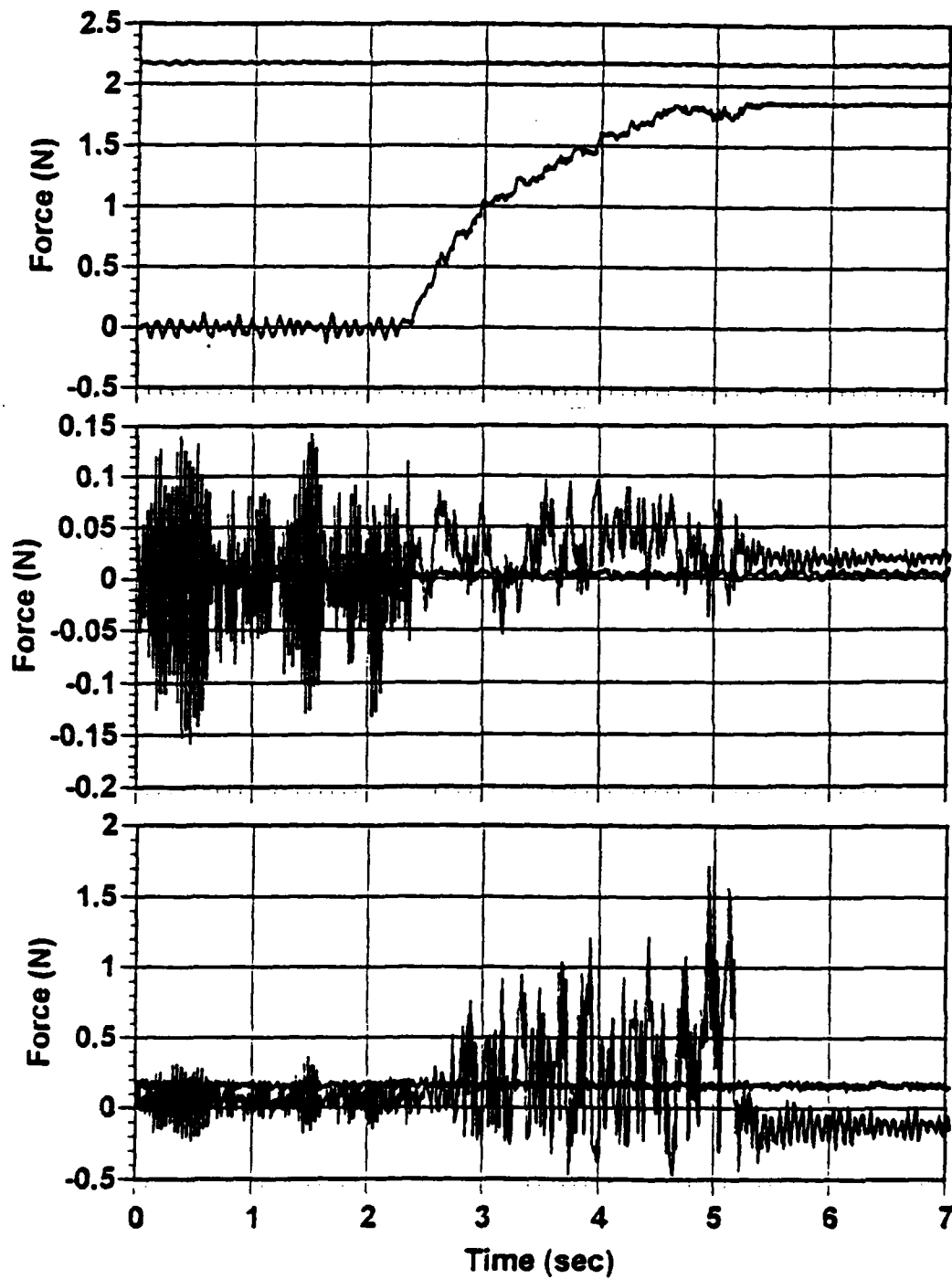
Figure 4.8.   Step Force Input in the x Direction, β=0.5.

58
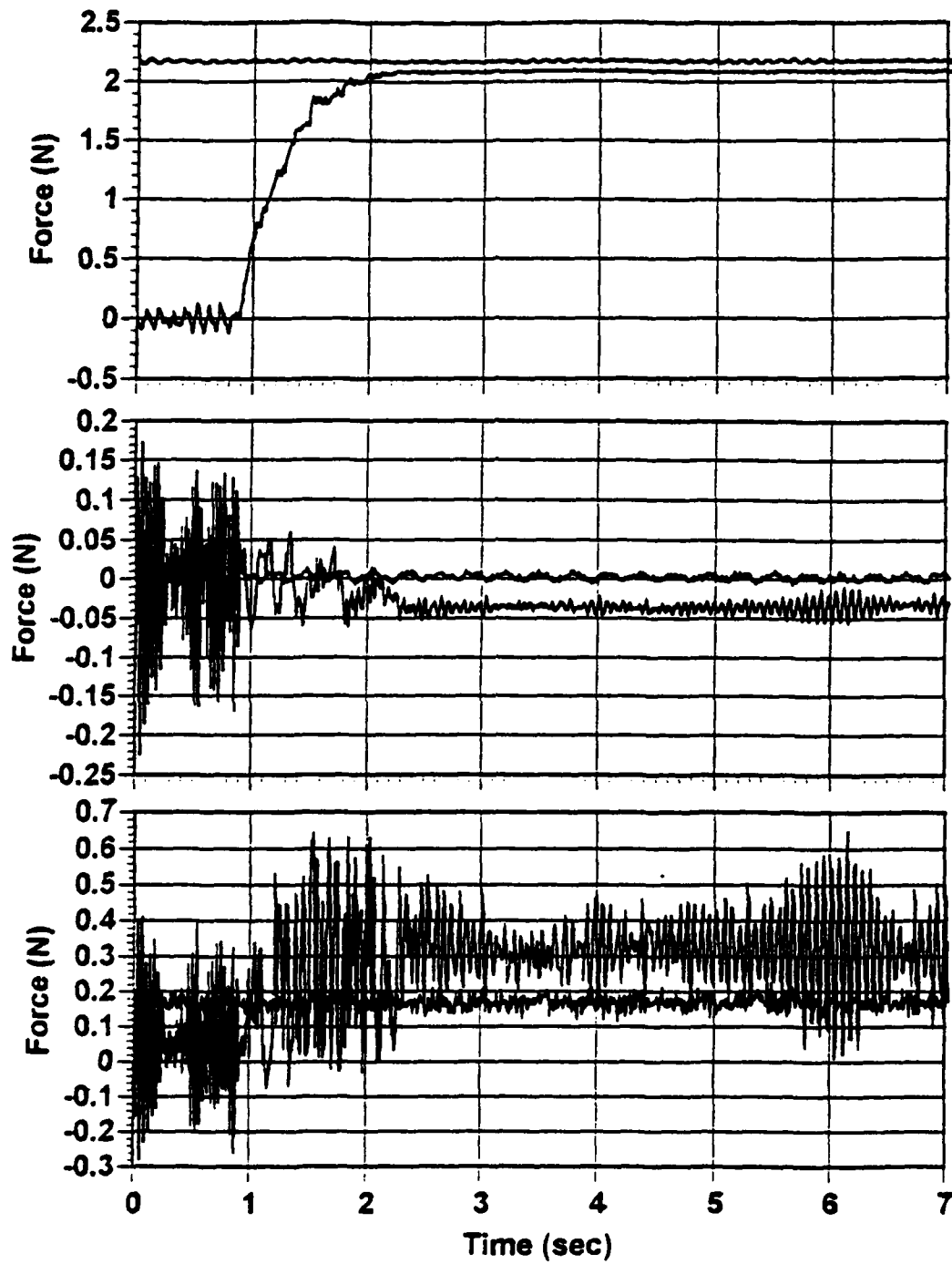
Figure 4.9. Step Force Input in the x Direction, β=1.
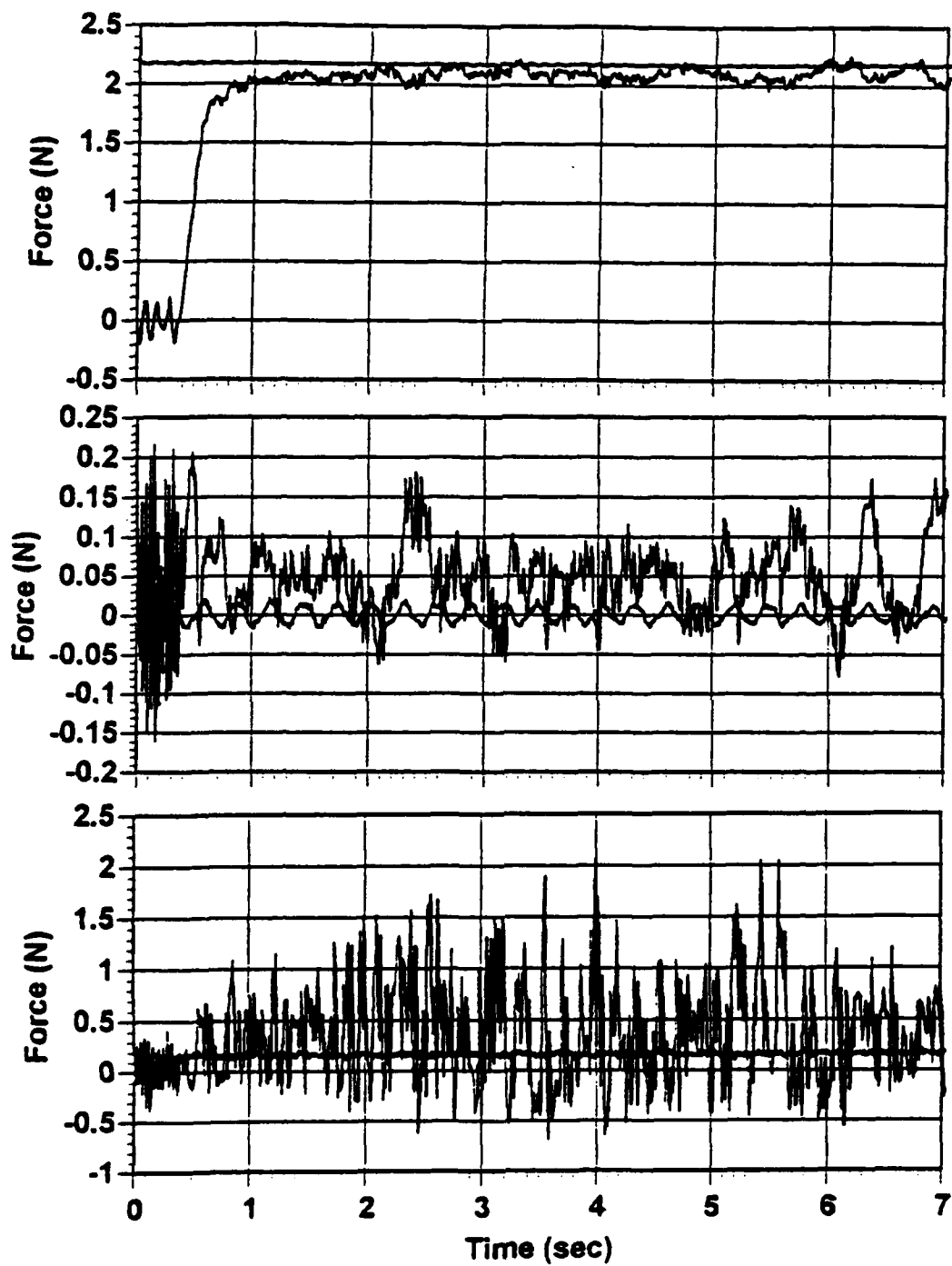
Figure 4.10. Step Force Input in the x Direction, β=2.
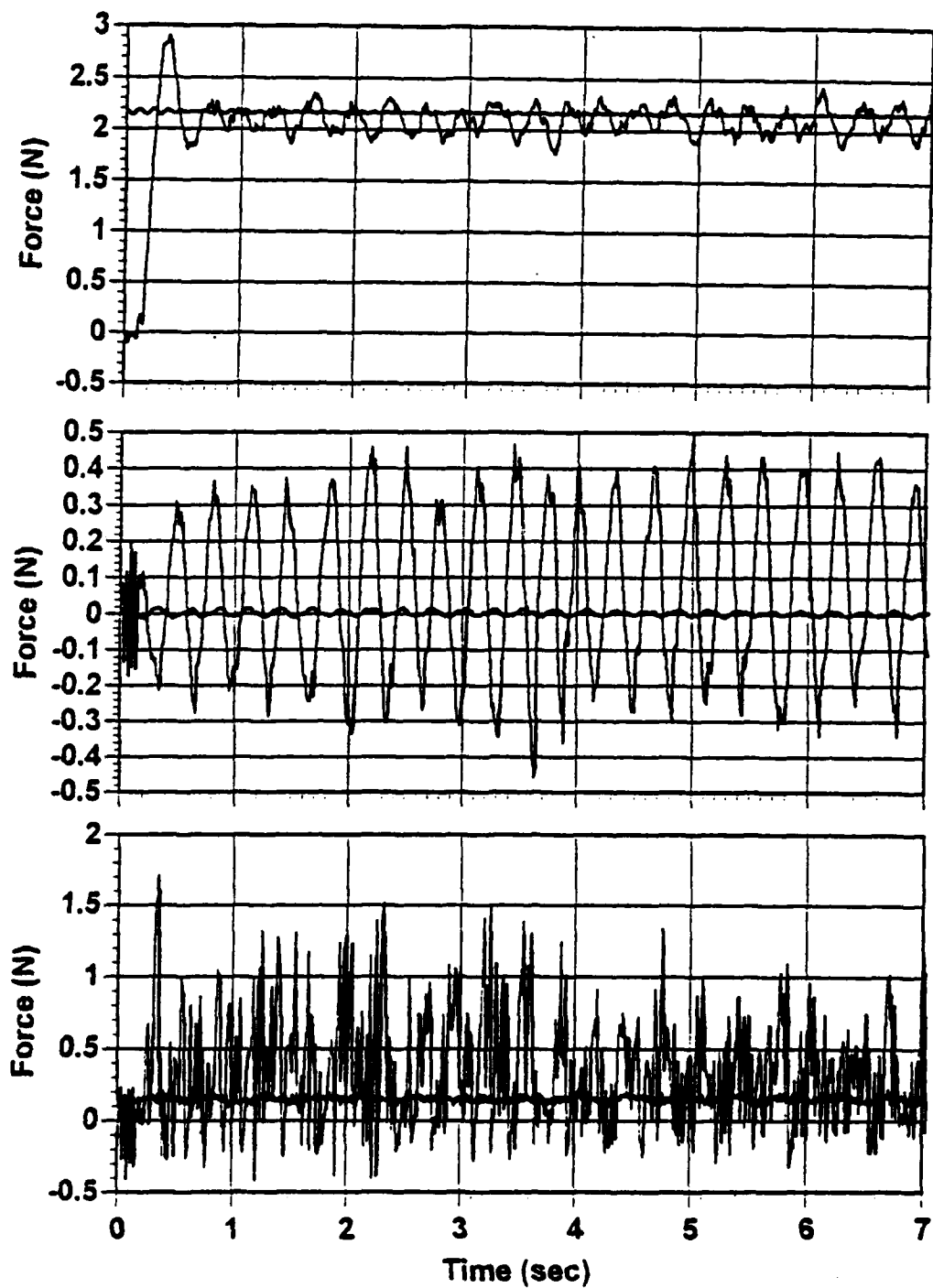
Figure 4.11.   Step Force Input in the x Direction, β=4.

61
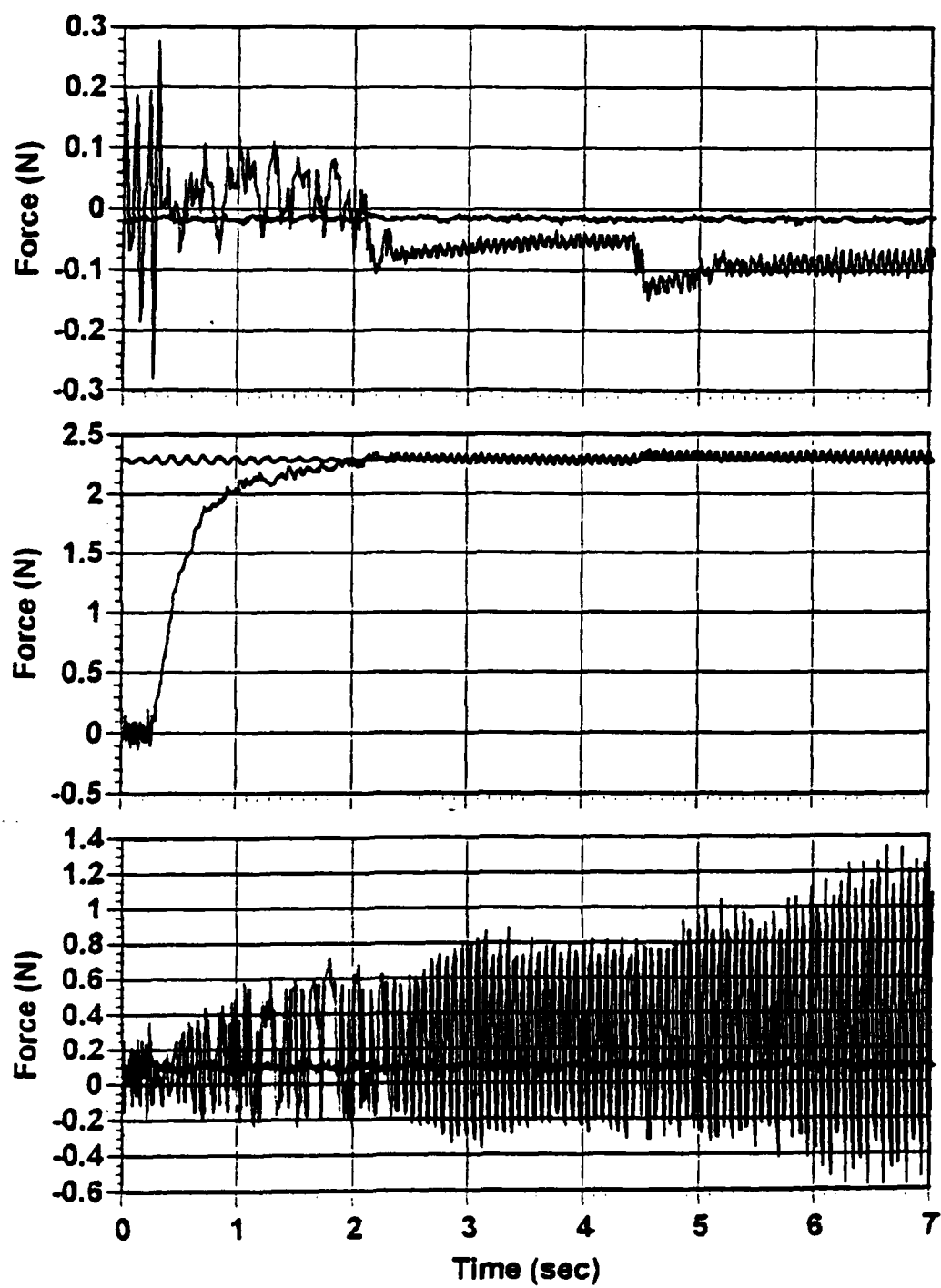
Figure 4.12.   Step Force Input in the y Direction, β=1.
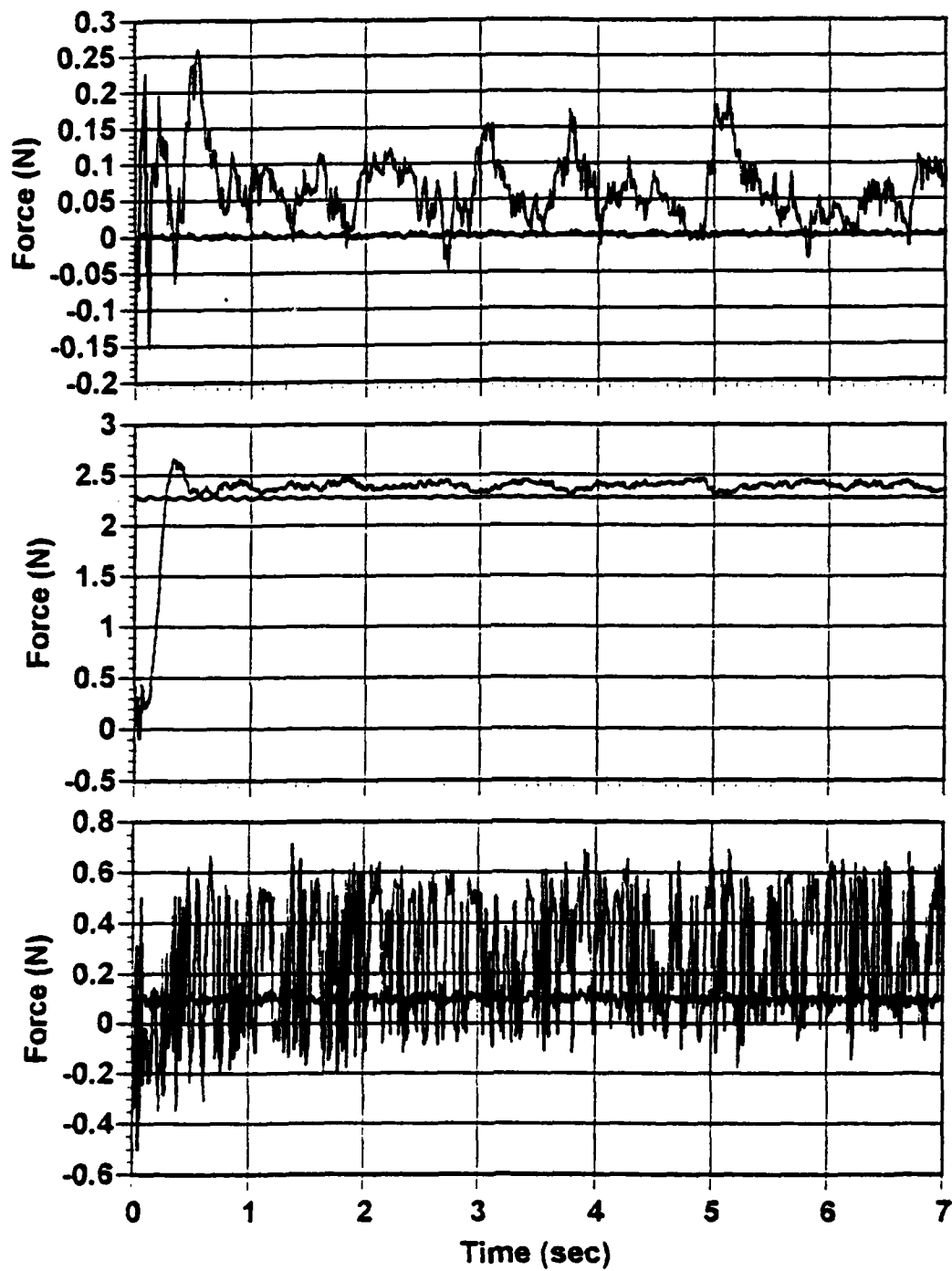
Figure 4.13.   Step Force Input in the y Direction, β=2.
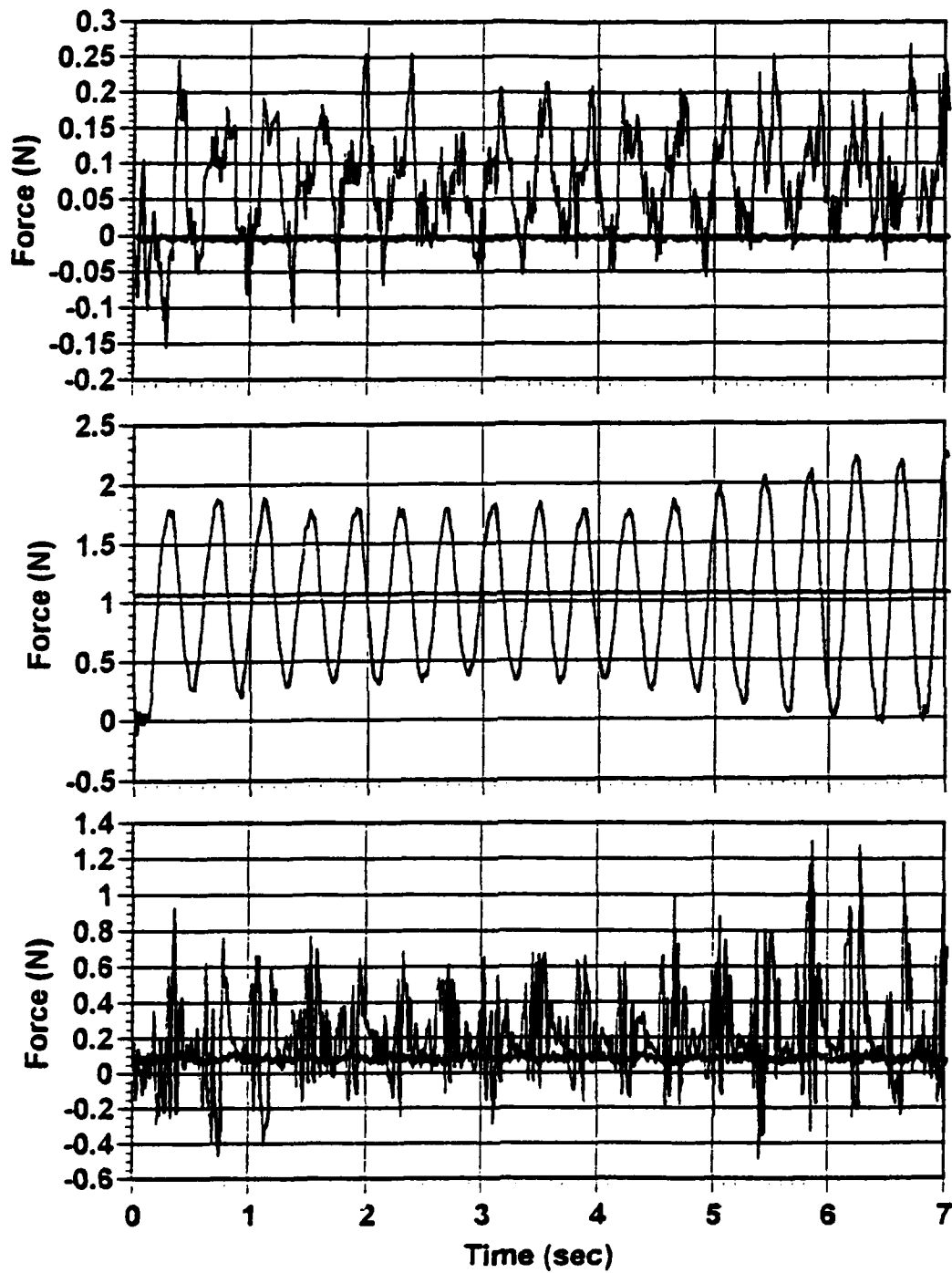
Figure 4.14. Step Force Input in the y Direction, β=4.

64
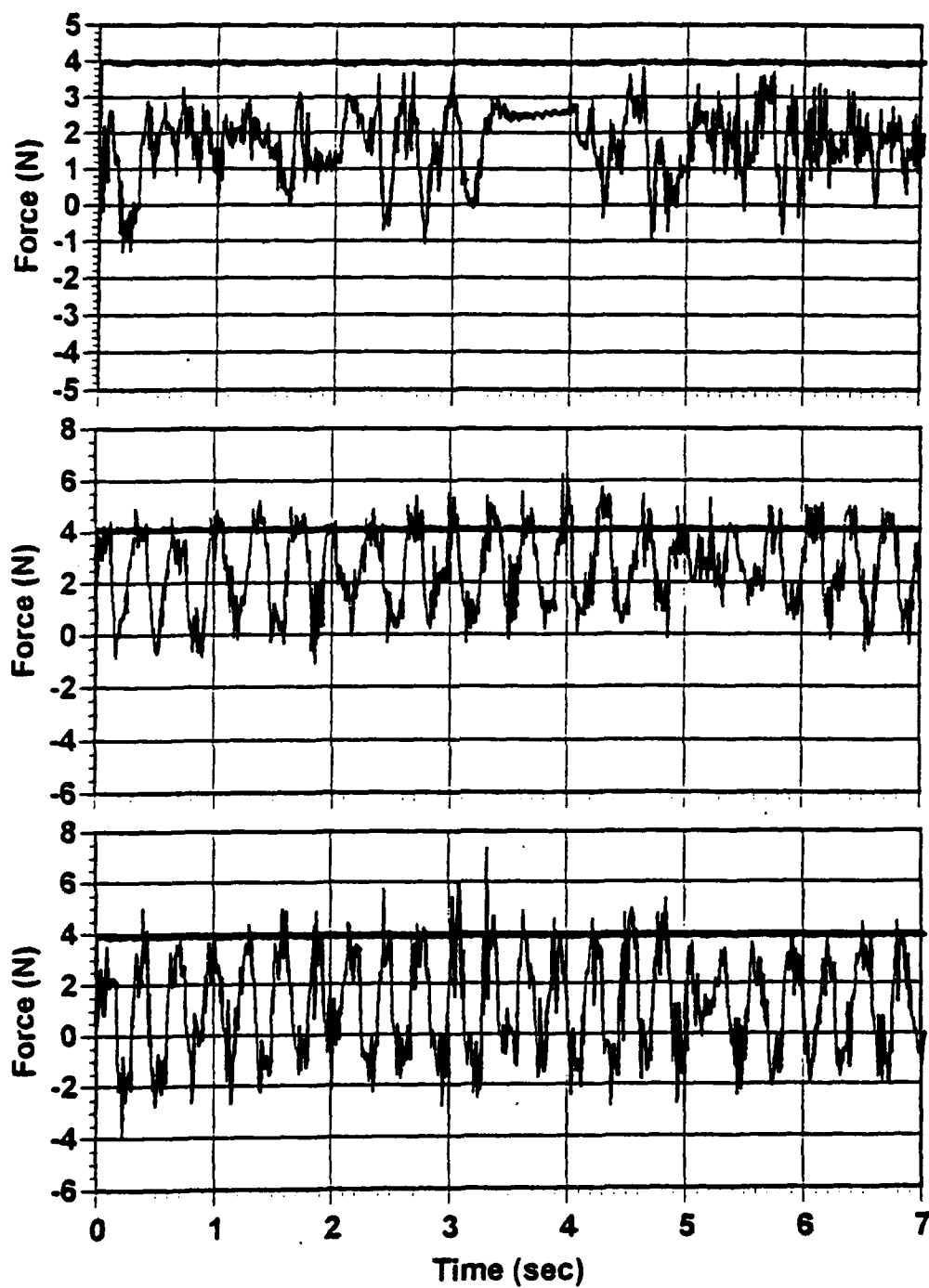
Figure 4.15. Step Force Input in the z Direction, β=20, β=40, and β=60.

Only the response in the z direction is shown for the input force in that direction and the compliant constraint is utilized.

Figures 4.16 and 4.17 show the response to a 4.4 N nominal force command in the x direction with $\beta=1$. Figure 4.16 uses PID control for the individual joint servos and Figure 4.17 uses PD control in the individual joint servos.

(2) Sinusoidal Input

Figure 4.18 shows the force response of the manipulator when subject to a sinusoidal force command in the x direction at a frequency of 0.33 Hertz and with $\beta=1$, $\beta=2$, and $\beta=4$, respectively. The top graph is for $\beta=1$, the middle graph is for $\beta=2$ and the bottom graph is for $\beta=4$. Only the response in the x direction is shown and integral control is used in the individual joint servo controllers. Figure 4.19 shows the response for the same force command and values of β, but with no integral control in the individual joint servo loops.

Figure 4.20 shows the response for a 1.25 Hertz force command and $\beta=1$ and $\beta=2$, with integral control in the individual joint servo loops. The top graph is for $\beta=1$ and the bottom graph is for $\beta=2$. Figure 4.21 shows the force response in each of the three directions for a 1.25 Hertz force input in the x direction with $\beta=2$ and no integral
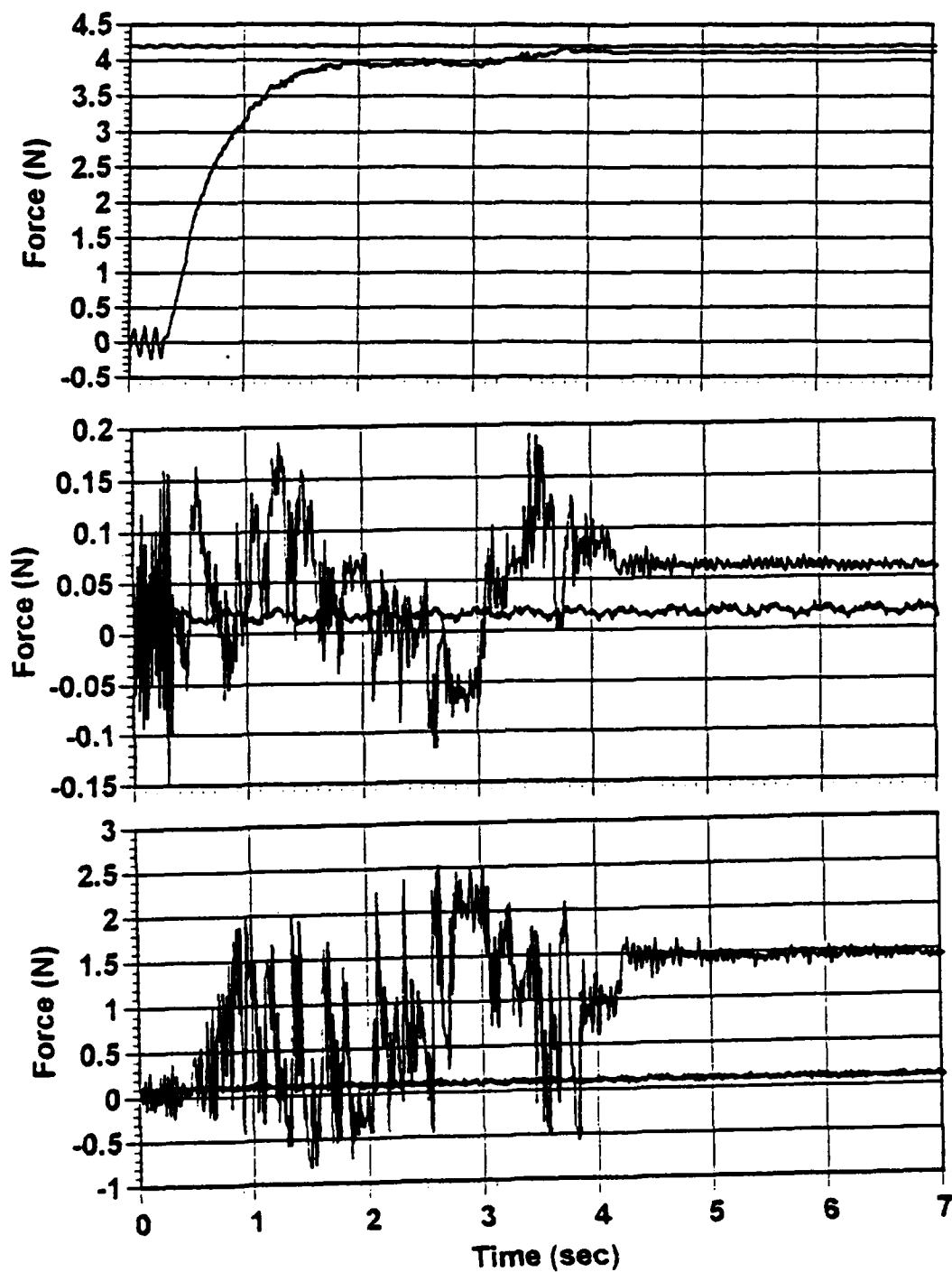
Figure 4.16. Step Force Input in the x Direction of 4.4 N,
β=1, and PID Control of Joint Servos.
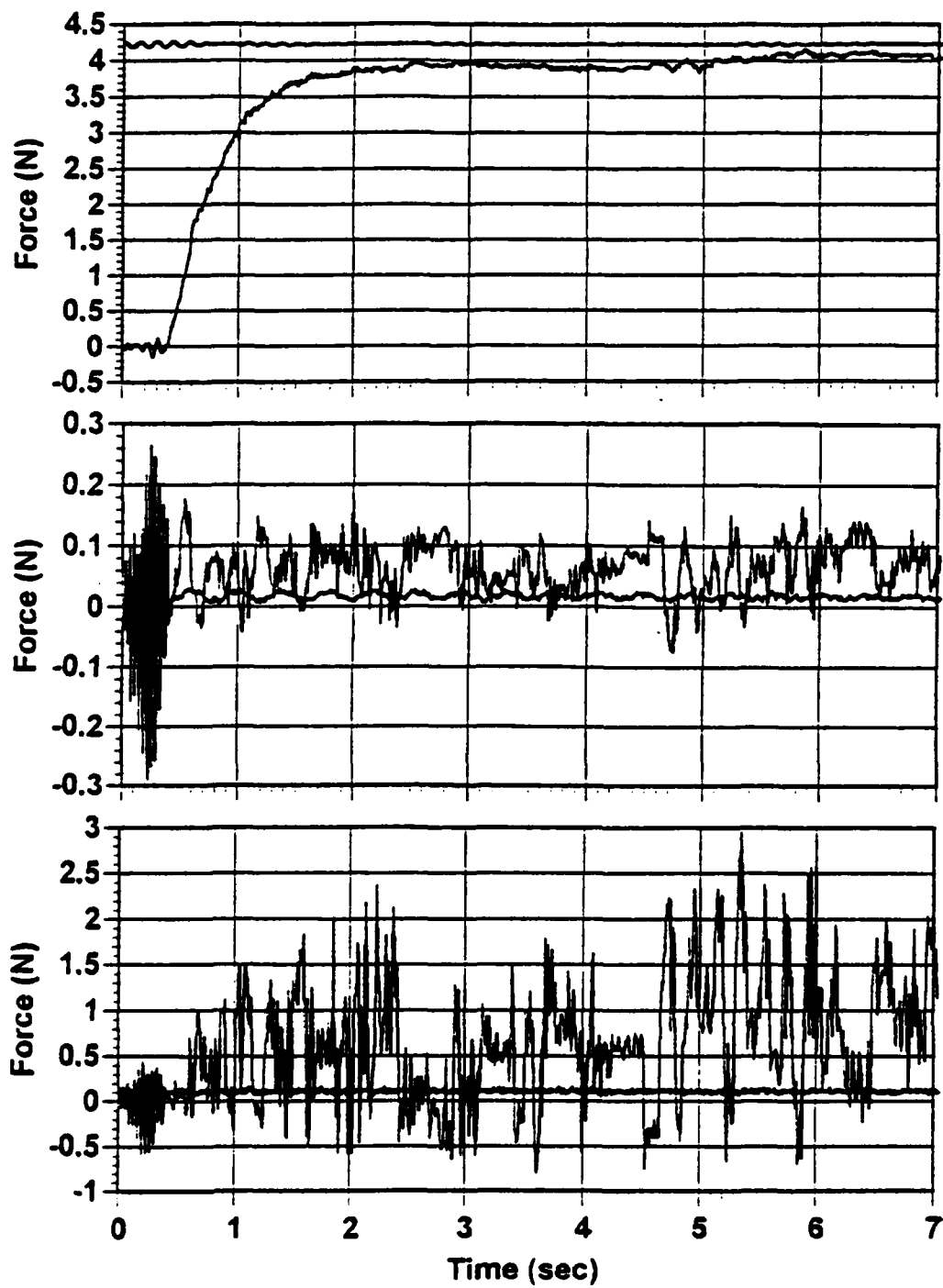
Figure 4.17. Step Force Input in the x Direction of 4.4 N, β=1, and PD Control of Joint Servos.
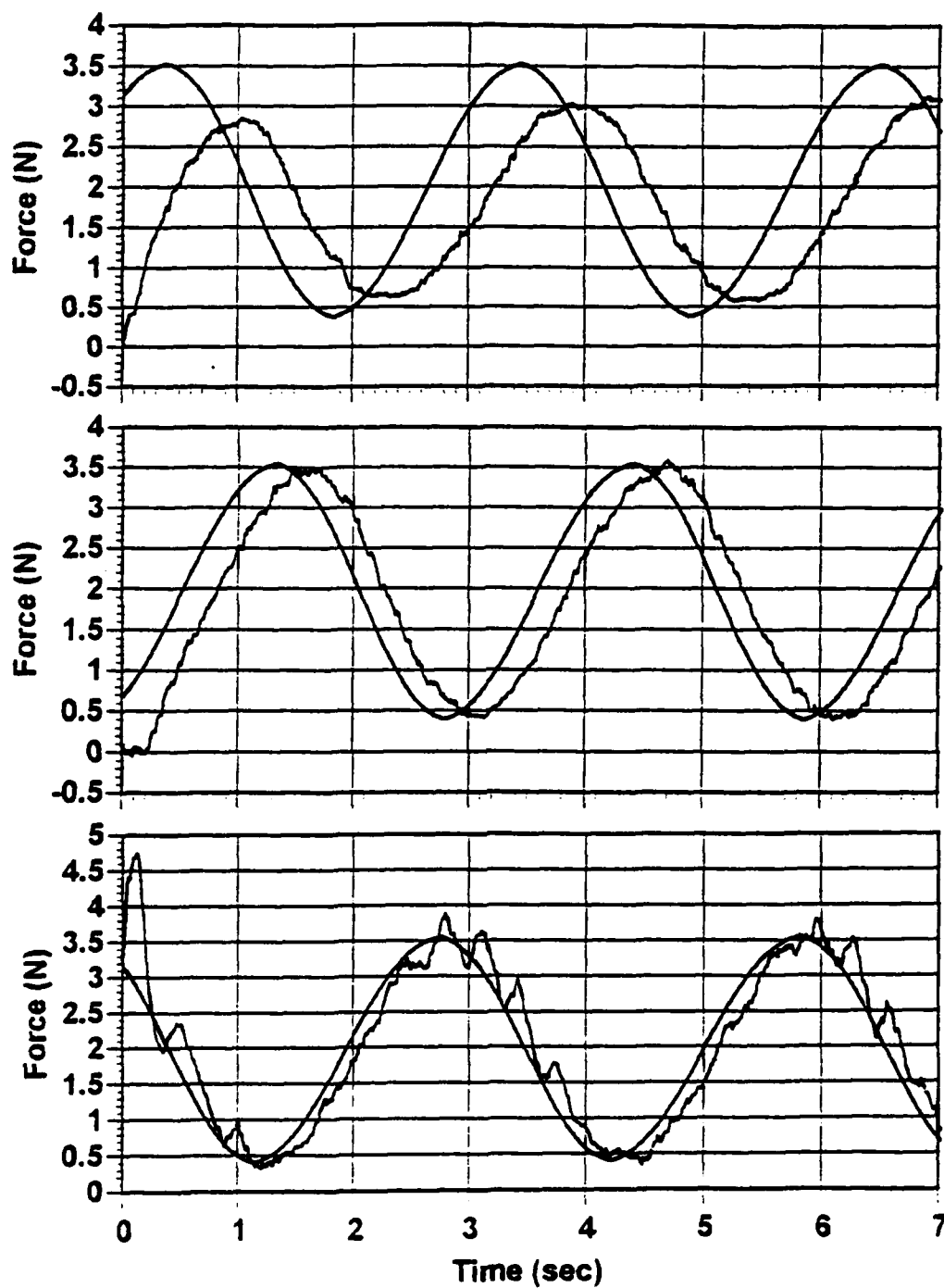
Figure 4.18. 0.3 Hz Force Input, β=1, 2, and 4, PID Control of Joint Servos.
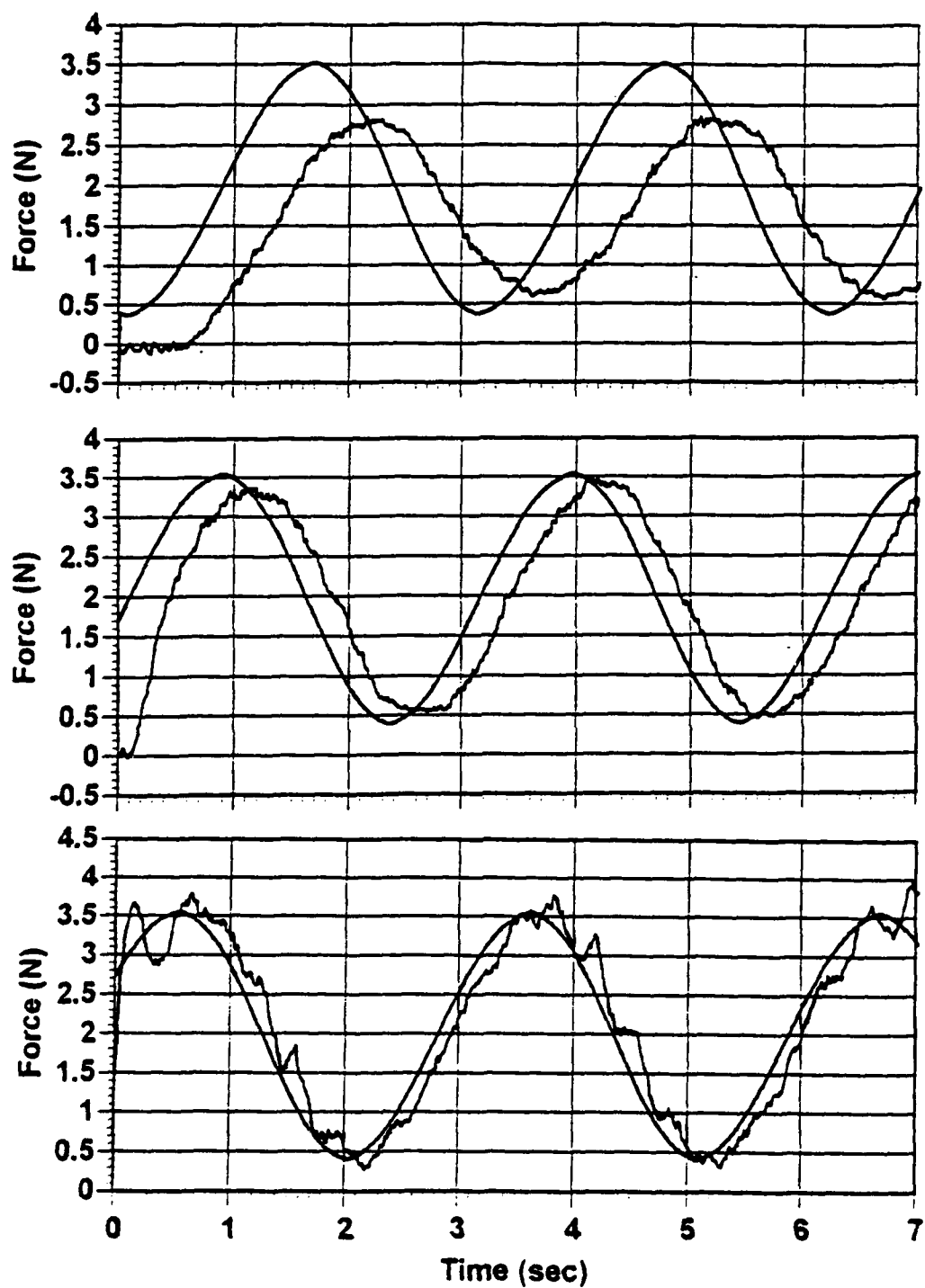
Figure 4.19. 0.3 Hz Force Input, β=1,2 and 4, PD Control of Joint Servos.
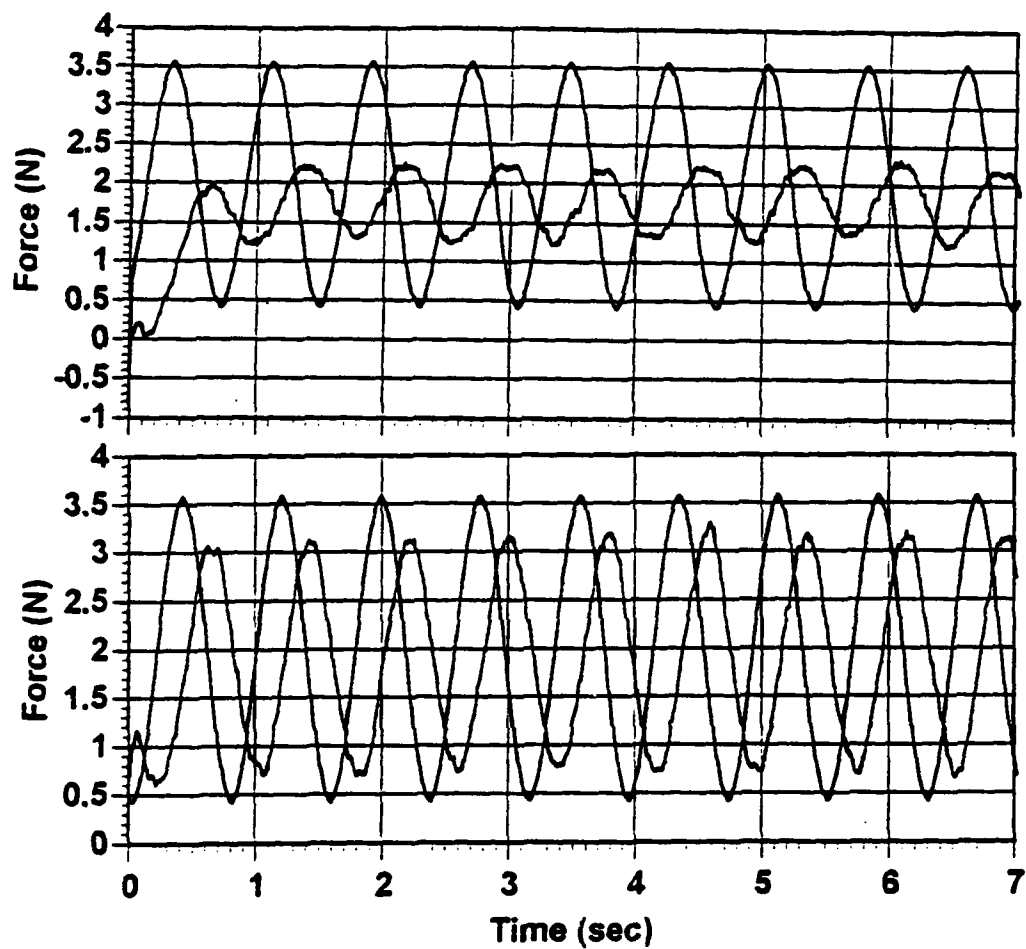
Figure 4.20. 1.25 Hz Force Input, β=1 and 2, PID Control of Joint Servos.
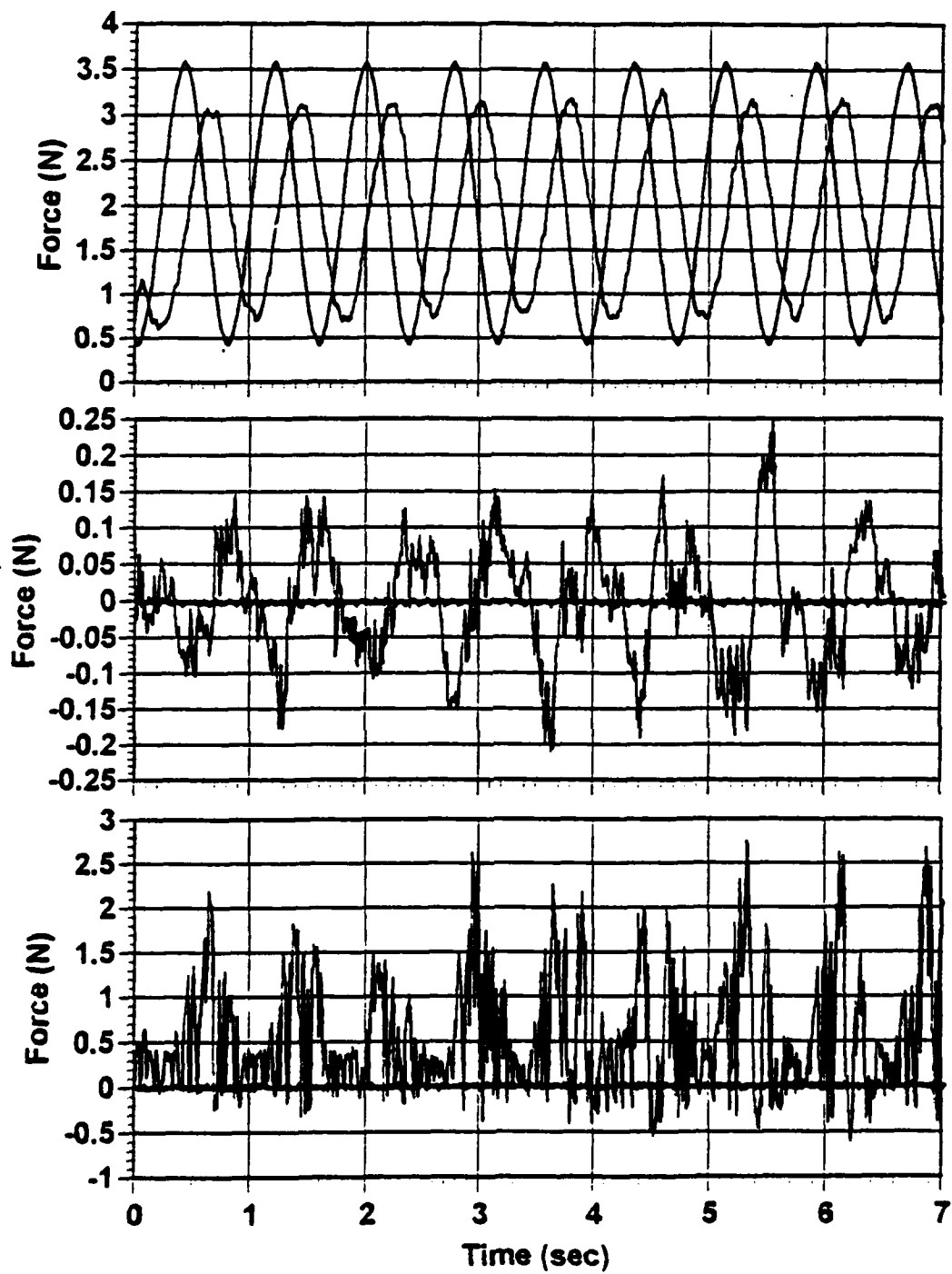
Figure 4.21. 1.25 Hz Force Input, β=2, PD Control of Joint Servos.

control in the joint servo control loops. As before, the top graph is the x direction, the middle graph is the y direction, and the bottom graph is the z direction.

2. Rate Control in Free Space

a. Description

In addition to the force response above, qualitative data is obtained from controlling the manipulator in free space. This allows for an evaluation the speed of response and the effectiveness of the control algorithm of producing the characteristics of a rate control system when no force constraints exist at the end effector. This also tests the ability to position the end effector at a desired location using the joystick.

The joystick is used to move the manipulator around in free space under rate control. A force applied to the joystick is varied in magnitude and direction and the motion of the manipulator is monitored. The values of **maxspeed** and **movetime** in the control program are varied to determine their effect on the response.

b. Results

The manipulator maintains a continuous path motion for values of **movetime** down to 40 msec in several manipulator orientations tested, with a **maxspeed** value of 462 mm/sec. With a constant value of **movetime** set of 60 msec, the value of **maxspeed** is lowered form 462 mm/sec until continuous path

motion stops and the manipulator's motion becomes jerky. The value of **maxspeed** at which this occurs varies depending on the orientation of the manipulator. Remember that **maxspeed** is an estimate of the maximum speed of the manipulator in cartesian coordinates which is used as a devisor in determining the distance for incremental motions in the control program. Continuous path motion is lost for values of **maxspeed** in the range of 200-300 mm/sec.

Using the same value of β in each direction resulted in varying rates of motion in each direction. The value of β must be raised considerably to achieve a reasonable rate of response in the z direction.

3. Task-Oriented Tests

a. Description

Tests are performed to demonstrate the ability to position the end effector against the single axis joystick utilized in the preliminary work described in chapter two. This single axis joystick serves as an environmental constraint. The manipulator's end effector is first positioned against the constraint and then a force is applied to attempt to reposition the constraint. This test is particularly representative of the type of tasks this system would be best suited for.

b. Results

The manipulator is positioned against the constraint with little difficulty and force is applied by the manipulator. The force input is slowly increased until the toggle switch resistance is overcome and the constraint is repositioned. Contact is lost between the manipulator and constraint during this motion as the constraint's springing action to the next position occurs rapidly. The manipulator then moves until it makes contact with the constraint again. Motion continues until forces are built up proportional to the commanded force applied at the joystick.

Once the constraint has been repositioned an additional force is then applied to the constraint. In this situation the manipulator slides around the spherically shaped constraint and the manipulator continues moving around the constraint. The manipulator actually slides around the constraint as depicted in Figure 4.22.

4. Control of Inertial Loads

a. Description

Tests are conducted to evaluate the performance of the system when handling an inertial load at the tip of the end effector. A small disk is placed on a flat, low friction surface in the manipulator's work space. The mass has a hole countersunk into it large enough to insert the tip of the end effector with considerable clearance. Tests are conducted to
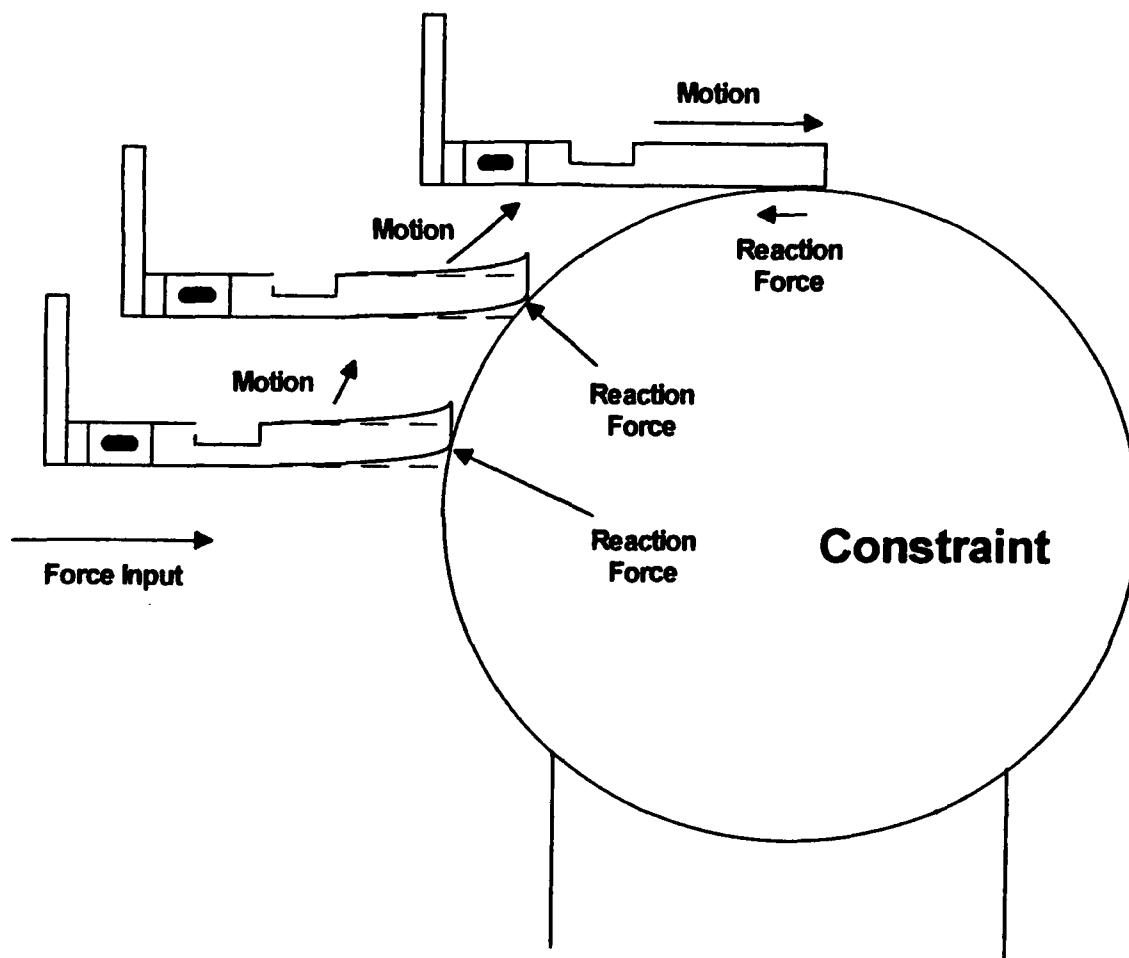
Figure 4.22. Response with a Spherical Constraint.

position the end effector in this recess and move the disk
around on the flat surface. This approximates the response of
handling an inertial load in two dimensions.

    b. Results

        The end effector is positioned into the hole of the
disk with little difficulty using the joystick as a rate
control input to the manipulator. Once contact is made

76

between the end effector and the disk a steady force/rate command is applied to the joystick. The disk and manipulator begin to move in the commanded direction but oscillate while moving. The disk slows down or stops moving momentarily during the oscillations.

C. SYSTEM SIMULATION

1. Description

A BASIC program is written to simulate the actual control structure of the PUMA 560 in a single DOF. The dynamics of the manipulator are assumed to be the critically damped or slightly underdamped system used for the system analysis of chapter two. A copy of the simulation program is provided in Appendix B. Figure 4.23 shows a flow diagram for the program.

The simulation algorithm consists of three loops. The inner most loop provides the simulation for the actual dynamics of a joint servo as described in the previous section. A time step of 0.875 msec is chosen since this is the update time of the PUMA 560's joint servo control loops. The next loop simulates the updated desired joint positions calculated every 28 msec by the main controller and commanded to each joint servo controller. The outermost loop simulates the updated joint position and velocity commands calculated and provided to the main controller from the VAL control program as discussed earlier in this chapter. A brief
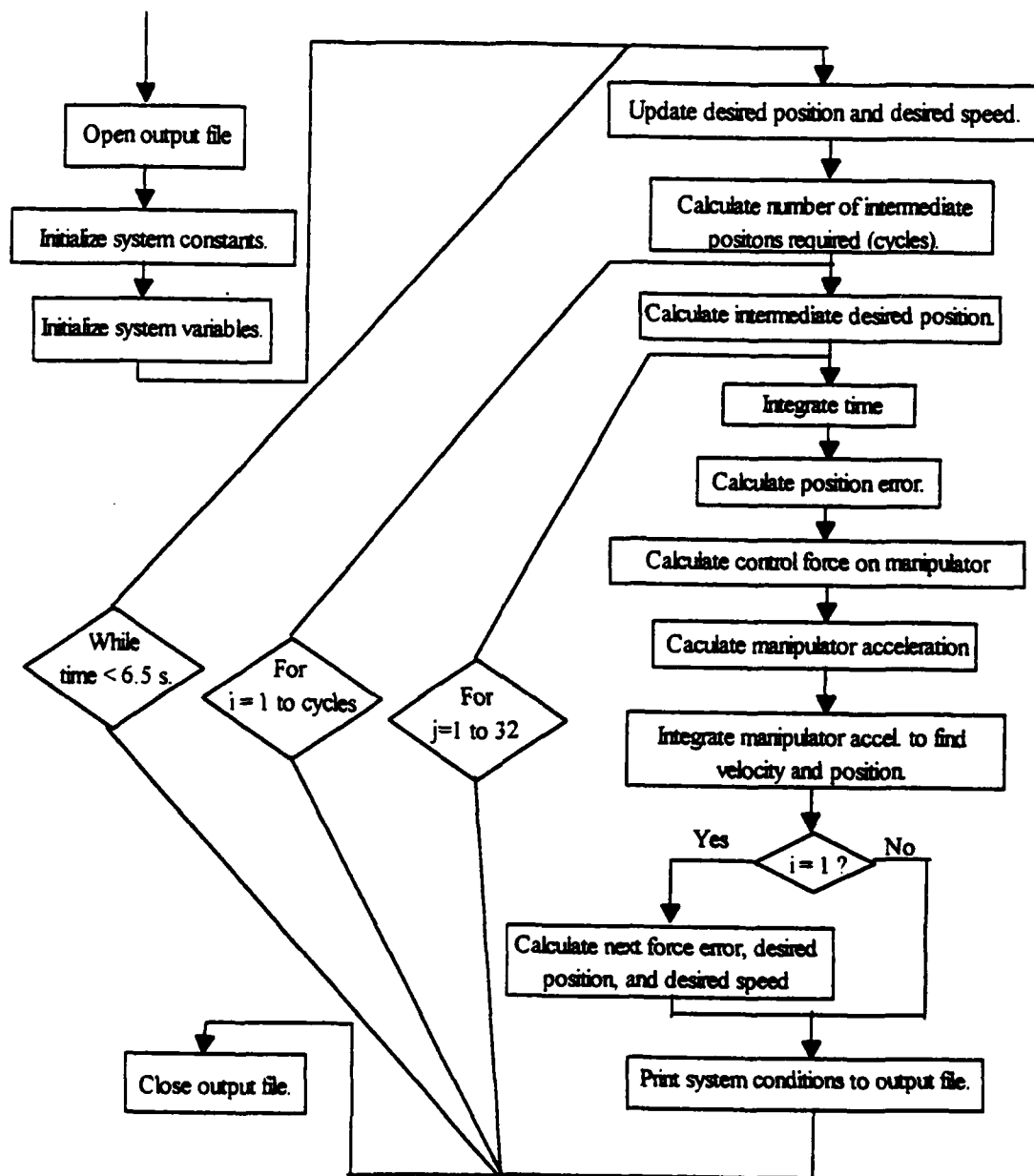
Figure 4.23. Simulation Program Flow Diagram.

78

description of each of the variables used in the program is listed at the beginning of the program.

A delay in processing new position and velocity commands is simulated within the outer two loops. The velocity command is based on performing a motion command in approximately 140 msec. This value was experimentally determined by adjusting this value and comparing simulation results to the actual response. The program requires a minimum of two 28 msec updates from the main controller to the joint servo controllers for each motion command. This simulates the minimum of approximately 60 msec between motion commands for continuous path motion. New force data, which the next motion command is based on in the control program, is estimated to occur at the end of the first 28 msec inner loop time period. This new motion command is not actually performed until the completion of the current motion command. This is designed to take one to two more 28 msec time periods. This effectively inserts a 28-56 msec computational time delay in the control structure.

2. Results

The simulation program is run using various values for the term β and estimates of system stiffness values and other control values. The effort is made to simulate the same conditions used in the experiments of the actual system described above.

Figure 4.24 shows the response for simulations with β=0.5, β=1.0, and β=2 for an end effector stiffness being the estimated stiffness for the x direction in actual system. The commanded force is a 2.2 N step input. Figure 4.25 shows the simulation responses with the same conditions but with the stiffness being the estimated stiffness in the y direction.

Figure 4.26 is the simulation response to a 0.3 Hz force input with β=0.5, β=1.0, and β=2 and an end effector stiffness being the estimated stiffness for the x direction in actual system. Figure 4.27 shows the simulations to a 1.25 Hz force input with the same system parameters. Figure 4.28 shows the simulations to a 1.25 Hertz force input with the same system parameters but with β=0.25.
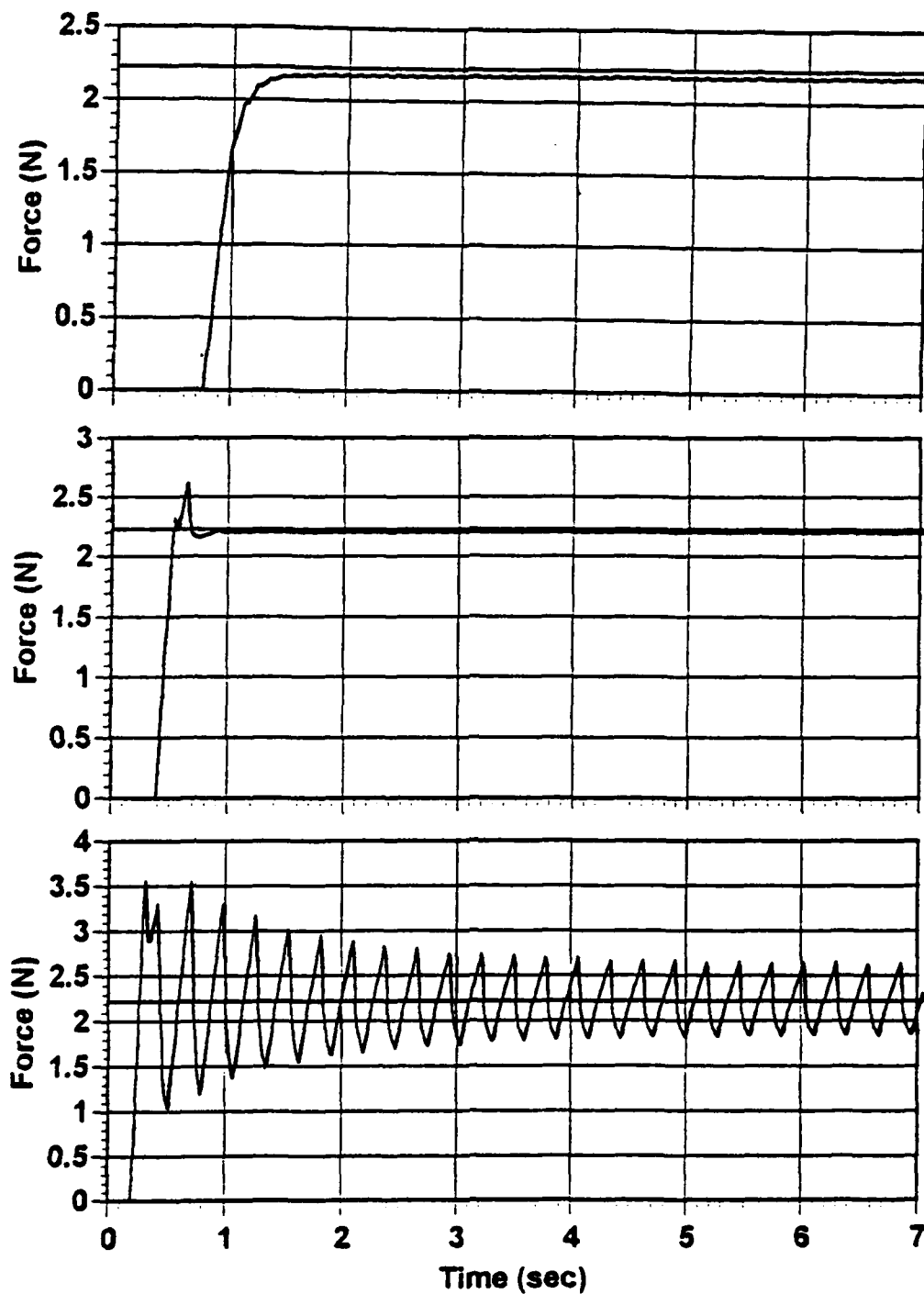
Figure 4.24. Simulation Response to Step Input in the x Direction, β=0.5, 1, and 2.

81
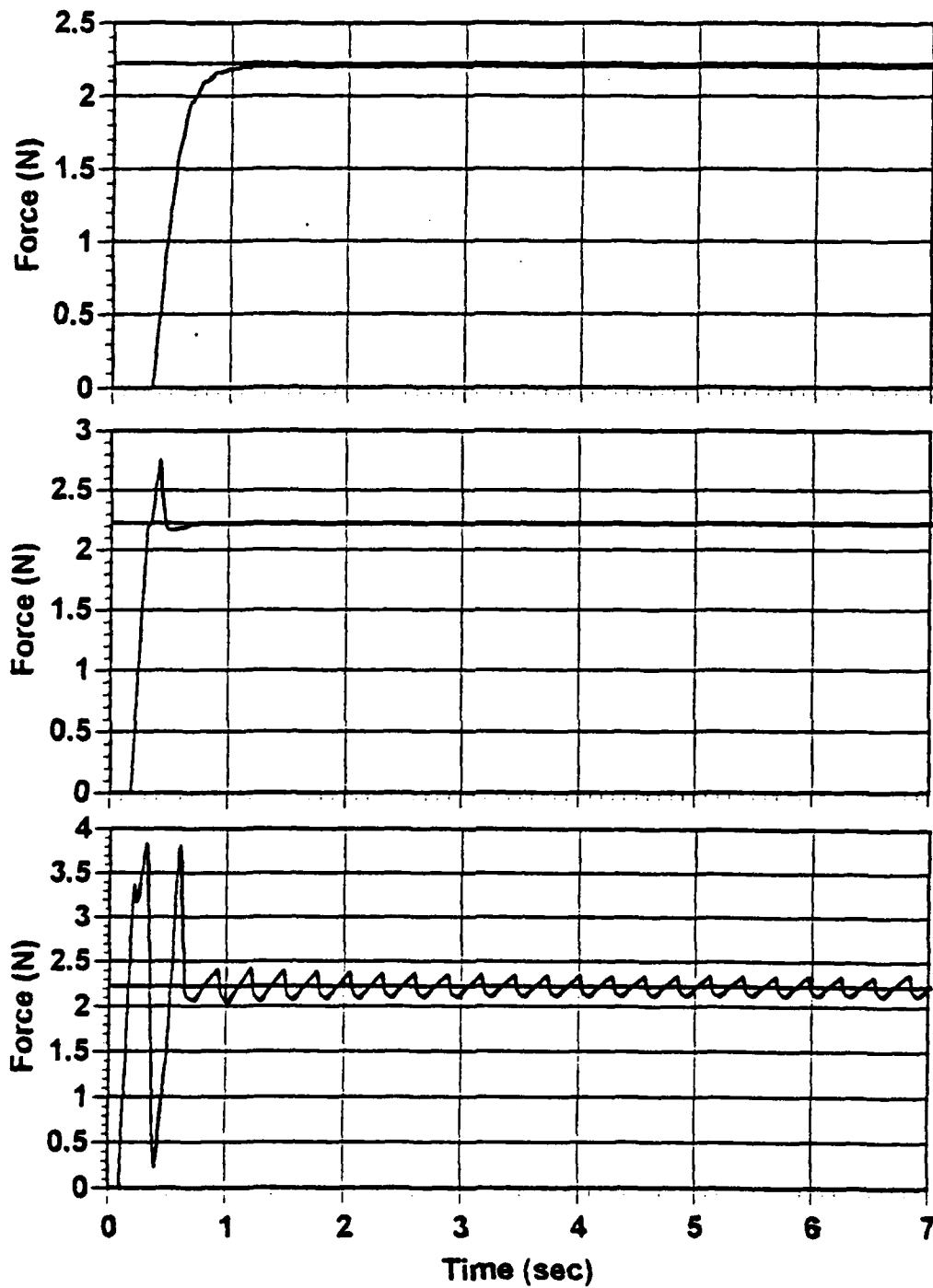
Figure 4.25. Simulation Response to a Step Input in the y Direction, β=0.5, 1, and 2.
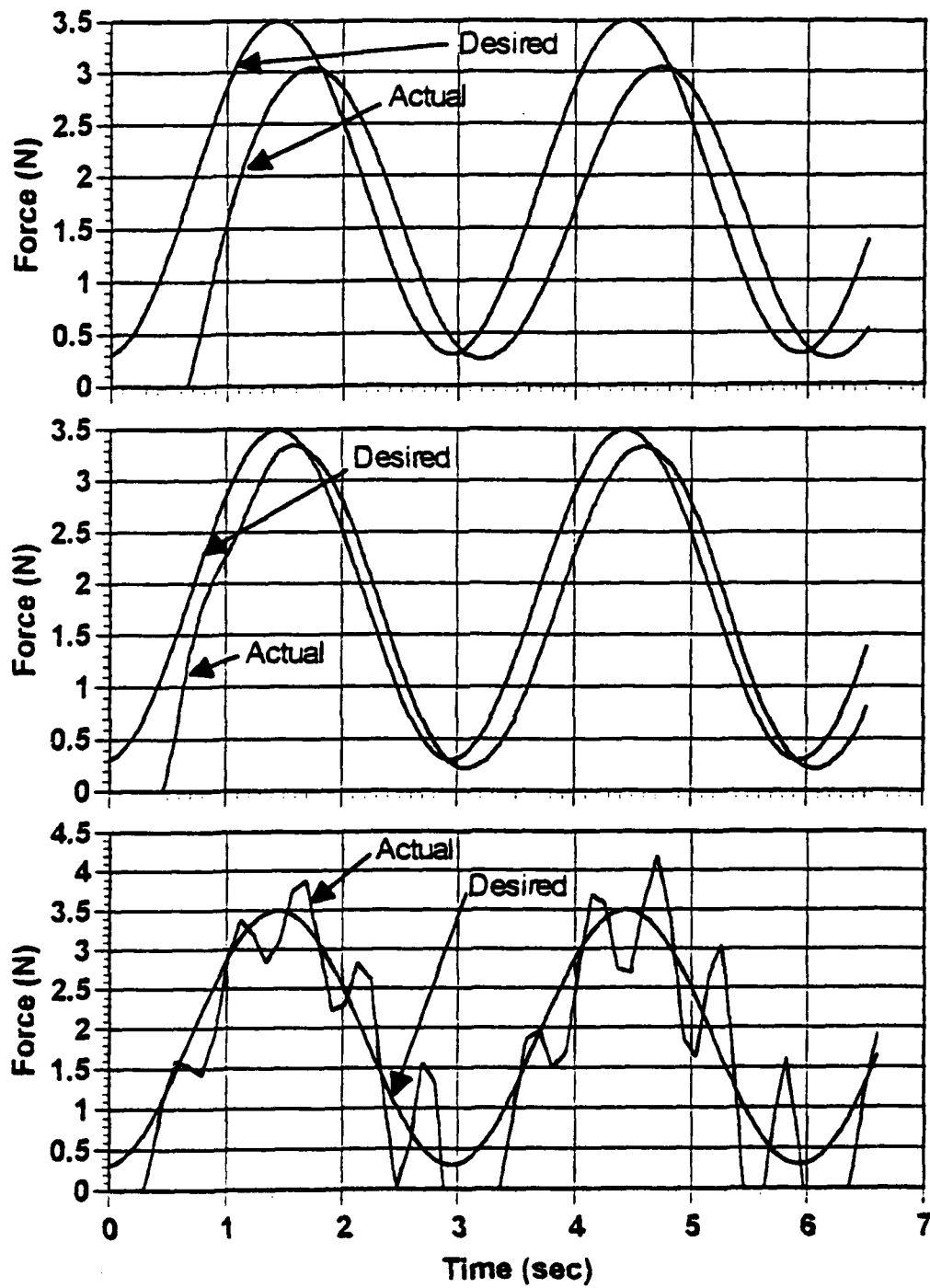
Figure 4.26. Simulation Response to a 0.3 Hz Input in the x Direction, β=0.5, 1, and 2.
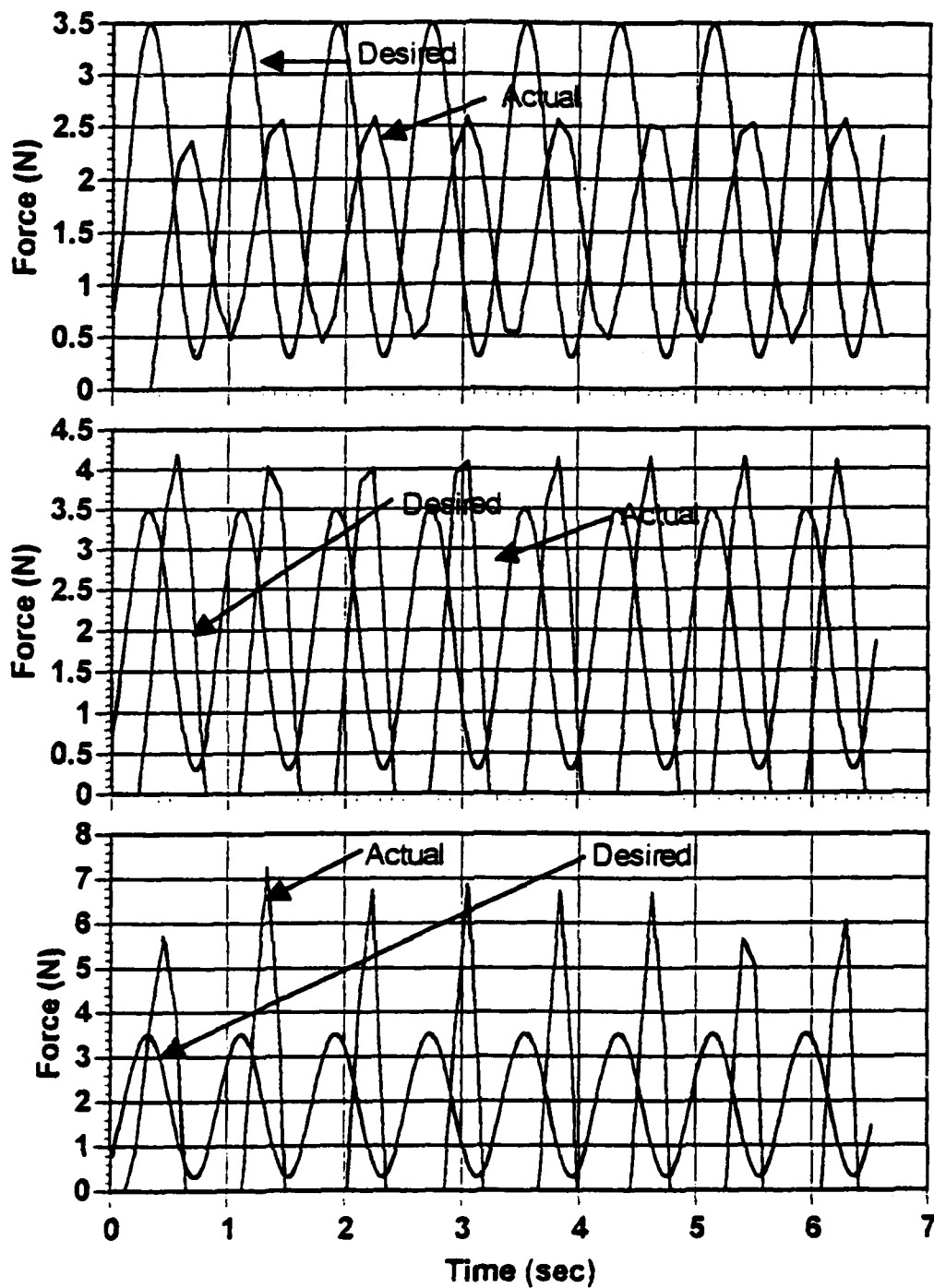
Figure 4.27. Simulation Response to a 1.25 Hz Input, β=0.5, 1, and 2.
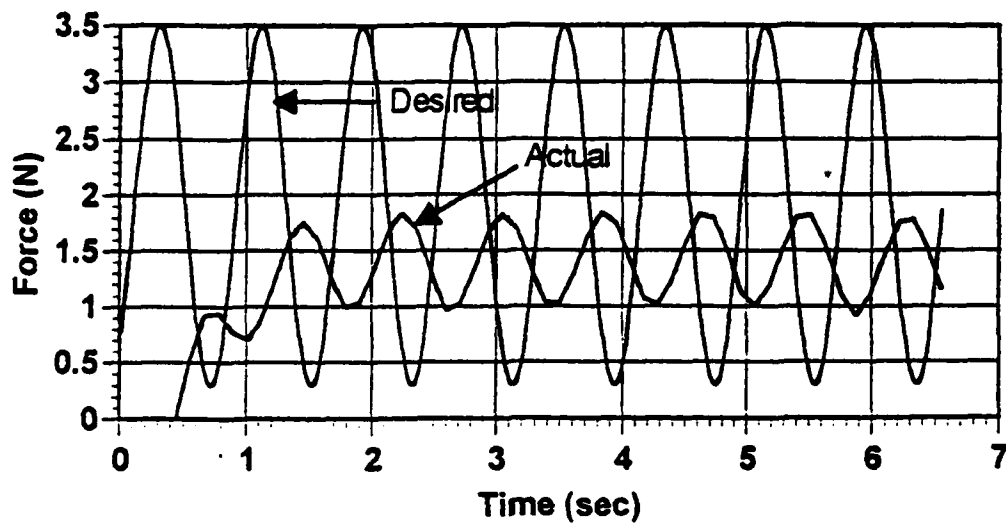
Figure 4.28. Simulation Response to a 1.25 Hz Force Input, β=0.25.

V. DISCUSSION

A. SYSTEM PERFORMANCE

1. Motion in Free Space

The manipulator behaves as expected during rate control tests in free space. The control program parameters of **movetime** and **maxspeed** can be adjusted to provide a quick response while maintaining continuous path motion. Without this continuous path motion capability in the PUMA control system, the system response is relatively poor. The capabilities of the control system of the manipulator greatly influence the ability to apply this force override rate control system to it. Other systems without continuous path control would not perform well with this control algorithm.

Using the same value of ß in each direction results in a varying rate of response in each direction for the same deflection of the joystick. This is due to the differing stiffness values for the end effector and joystick in each direction. Difficult control of the manipulator in some situations results, especially when trying to move the end effector in the z direction. The high stiffness of the end effector in this direction, combined with the coupling between the strain gages measuring forces in the y and z directions, makes motion control in the z direction very difficult. Using

an end effector and joystick with the same stiffness in all directions would improve controllability in this respect. Additionally, reducing the amount of coupling between force sensors in each direction would also improve performance. These two issues must be considered when designing and constructing a new end effector and joystick for the system.

As mentioned in the last chapter, the rate of the manipulator's motion depends on the orientation of the robot during the motion and the direction of the motion. The rate commands in the control program provide a relative speed of motion in joint space, not cartesian space. This change in the rate of motion is clearly noticeable during motion control but does not adversely effect the ability to position the manipulator. An updated version of the VAL programming language allows for control of the speed in cartesian space when performing straight line motion commands. This would alleviate this problem of varying speed but the delays associated with the additional computations required for the straight line motion would adversely effect the stability of the system.

Efforts to control the system with integral control of the force error signals does result in motion that is not characteristic of rate control but more like acceleration control. Difficulty in stopping the manipulator at a desired location, or stopping it at all, becomes quite difficult and

renders this type of control unacceptable for motion control in free space.

2. Response with Stationary Constraint

a. Step Force Input

The response is essentially that of a critically damped second order system. Although the force response does not indicate the manipulator's **motion** directly, the manipulator initially accelerates to a constant speed proportional to the force error between the joystick and end effector. Noticeable force oscillations occur in each direction as the manipulator moves. This is shown in Figures 4.8 through 4.12, 4.18, and 4.19. The frequency of these oscillations decrease considerably after contact is made with the constraint. This oscillation is due to the vibration of the end effector as the manipulator moves. Once contact is made with the constraint the end effector acts as a cantilever beam with a very large mass at the end. The natural frequency of the end effector in contact with the constraint decreases considerably. Some fluctuations occur in the commanded force also, as observed in Figures 4.9, 4.11, 4.12, and 4.19. This is due to some vibration of the joystick resulting from machinery operating near by.

Once contact is made, some forces are developed in the directions other than the direction of the commanded force. This can be seen in Figures 4.8 through 4.14. For

88

commanded forces in the x direction the force that develops in the y direction is very small and probably results from slight errors in the alignment of the constraint surface and the manipulator. If the rigid constraint is not perpendicular to the direction of the commanded force, the motion of the end effector results in a deflection of the end effector in the other directions and a resulting frictional forces in those directions (Figures 4.9 and 4.11). For a commanded force in the x direction and low friction between the end effector and the constraint, force errors in the y direction will result in motion of the manipulator at rate proportional to the force error in that direction. The end effector slides along the constraint surface with a constant reaction force between the end effector and the surface of the constraint. The same explanation describes the response observed in the z direction. However, much larger forces are expected to develop in the z direction due to the higher stiffness of the end effector in that direction.

Misalignment of the end effector's coordinate frame and the manipulator's tool frame may also contribute to this problem. For example, if the x and y coordinate frames of the end effector are not properly aligned with the x and y coordinates of the manipulator tool frame motion of the manipulator along the tool frame's x axis when it is against a constraint will result in a deflection of the end effector and measured forces in both the x and y directions. This

results in force errors in both directions and motion in both directions to try to alleviate the associated force errors. This is probably the source of the oscillations observed in Figure 4.11 in the y direction. This contribution to the force errors can be minimized with accurate alignment of the end effector and manipulator's tool frame.

The steady state response differs from that expected, with the steady state errors generally being lower than anticipated based on the analysis of Chapters II and III. Compare the Bode plot of Figure 2.9 to the responses shown in Figures 4.8 through 4.14. This result can be attributed to the discrete nature of the motion commands in the control program. Each motion command executed within the control loop can be looked at as a separate motion control command. The steady state error will depend most on the last motion command executed and the actual magnitude of the force command. With a larger force command larger forces develop at the end effector. This translates to a larger disturbance on each of the joint servos opposing motion in the desired direction. The results support this discussion.

The response is more rapid and a smaller steady state error results as the value of β increases(Figures 4.8 - 4.11, 4.12 and 4.13). This is true until β reaches a value of four where unstable behavior begins to be observed (Figures 4.11 and 4.14). The results are very similar for commanded forces in both the x and y directions, with the system

approaching unstable behavior for values of β between two and four. The response in the y direction is more rapid than in the x direction due to the lower stiffness of the end effector in the y direction. Compare the response rate of Figures 4.10 through 4.11 to those of Figures 4.12 through 4.14. This lower stiffness in the y direction corresponds to a higher force error gain for the same value of β. The stiffness is almost twice as high in the x direction, resulting in a response that is only about half as fast as the response in the y direction. From the results shown in Figures 4.9 and 4.12, one can see that the steady state error is much lower for the response to a commanded force in the y direction with the same value of β. This tends to indicate that the steady state error is not really a function of the value of β, but rather is dependent on the stiffness of the system, including the end effector, and the actual control system gains. This opposes the steady state error of the linear system of Equation 2.17. The results do confirm the advantages a system with lower stiffness has in controlling forces.

There is essentially no change in the speed of response and there is a minimal improvement in the steady state response when using integral control in the individual joint servo controllers. Compare the response of Figure 4.18 to that of Figure 4.19. This small gain in performance does not justify the risk of overloading the individual joint servos when integral control is used.

Adding integral control to the force control algorithm also fails to confirm any significant improvement in the steady state force error. Results of tests for which integral control to the force error is added did indicate that the concept works but the problems associated with control of motion in free space renders this control system approach inappropriate for this system. If the actual forces at the end effector are known, rather than the force errors between the end effector and the joystick, an adaptive control algorithm could switch on or off the integral control of the force error depending on whether contact is made with an obstruction or not. This approach would suit control of forces with a stationary constraint well, but would not be suited for control of inertial loads. The problem of accelerating the payload load motion rather than pure rate control would once again, result in this situation.

b. *Sinusoidal Force Input*

The results of Figures 4.18 and 4.20 indicate that response magnitude decreases and phase lag increases with the higher frequency input as expected based on the frequency response of the single DOF analysis of Chapter II. The response to a force command is measured for only two frequencies, but a more complete frequency response for a particular set of control gains and environmental constraints can be obtained by measuring the response for a complete

series of frequencies and plotting the results in the form of a Bode plot. This is not conducted since the system is expected to be modified by replacing the joystick and end effector with devices utilizing Force Sensing Resistors to measure forces. This will alter the response of the system and invalidate specific results found during these tests.

Figure 4.21 shows noticeable oscillations occurring in the forces developed in both the y and z directions at approximately the same frequency as the commanded force in the x direction. This is likely due to the errors in the alignment of the coordinate axes of the end effector and manipulator tool frames as discussed earlier. The magnitude of the oscillations in the y direction are relatively low while those of the z direction are much higher. This is expected due to the variation in stiffness of the end effector between these directions as discussed earlier. Notice that the response of Figures 4.18 and 4.19 has much less lag and a magnitude almost reaching the commanded force at the gain condition of $\beta=2$. The lag is even more reduced for $\beta=4$, but the system is approaching unstable behavior. The delays associated with program computations and data exchange clearly reduce the stability of the system.

3. Task-oriented Tests

The results of the efforts to move a constrained object around through contact forces provided unanticipated

93

results. The motion of the manipulator actually sliding around the obstruction is not expected but is easily explained. When the contact is made against the non-flat surface of the obstruction forces develop in directions other than the commanded direction. These forces can be resolved into normal and tangential components relative to the obstructions surface. They can also be resolved into the three directions of the end effector's reference frame. Force error signals in directions other than the commanded force direction develop, causing motion control commands and movement of the manipulator to reduce these force errors.

Several things can limit or prevent this type of response. If the end effector contacts the constraint in a way in which the reaction force is directly in line with the commanded force, no force signals will develop at the end effector in another direction to cause this lateral motion around the constraint. Also, a slight misalignment can occur as long as static friction between the end effector and the constraint is large enough to counteract the force error tangent to the constraint's surface. Another way to prevent the lateral motion is to reduce the control gains or set them equal to zero in the directions motion is not desired to prevent this lateral motion. This has the potential for allowing high forces to develop in these directions.

In many situations this accommodating motion is desired. For example, in the assembly task of inserting a peg

94

in a hole the lateral motion to center the peg in the hole as it is inserted is desired and this control system would automatically perform this accommodating motion. Forces in lateral directions the peg would be minimized.

This control system has the capability to be applied to do such tasks as washing a window or rotating a crank, tasks normally seen performed with a hybrid force/position controller. The system can be programmed to maintain a specified force normal to the window and then the rate commands can be issued using the joystick to move the manipulator along the surface since the motion is not constrained in direction parallel to the surface.

4. Control of an Inertial Load on a Flat Surface

The friction forces between the disk and the surface provide limited simulation of an inertial load but this oscillatory motion is an important response that would be expected when handling a purely inertial load. Normally the end effector itself has a very high natural frequency and vibration of the end effector is of small magnitude. This has a limited effect on the system response. In this situation, where a load has been added to the tip of the end effector, essentially, the frequency of oscillation decreases considerably while the magnitude increases. This results in an oscillations in the control system force error command

95

input which leads to oscillations in the manipulator's motion. This can result in an unstable response of the system.

A major factor contributing to this behavior while controlling an inertial load is the stiffness of the manipulator and end effector. In this application the stiffness of the manipulator is set by the PUMA control system and is considered unchangeable. This limits one to looking at the design of the end effector to prevent this undesired response. High mechanical stiffness of the end effector is desired to prevent this potentially unstable motion with an inertial load. This raises the frequency of oscillation and more importantly reduces the magnitude of the oscillation. Including some type of damper would also serve to limit the amount of oscillation which occurs when controlling an inertial load. This need for high stiffness in controlling an inertial load contradicts the desire for low stiffness to provide a rapid, stable response. A trade-off is required for unless the control system of the manipulator itself is changed to allow for variable control gains to change the manipulator's stiffness for an optimum response.

B. SYSTEM STABILITY

The test results demonstrate the limited stability of the system and the effect changing the control gains and the environmental constraints has on stability. It is clear from

the test results that there is a trade off between the system's speed of response and stability.

1. Stiffness of End Effector and Environment

The stiffness of the end effector, and of the manipulator, greatly influence the performance and stability of the system. The results indicate a correlation between the value of β and system stability exists. In both the x and y directions the system response becomes oscillatory at $\beta=4$ when subjected to a step input. Oscillation occurs in the x direction when subjected to a sinusoidal input at this same value of β. See Figures 4.11 and 4.14. This occurs for a rigid environmental constraint. A similar correlation can be developed for the effect changes in environmental stiffness has on stability. Oscillation appear to occur at the same value of β regardless of the magnitude of the actual force command. This indicates no connection between the magnitude of the input and the system stability.

Tests can be run to determine the limiting value of β for maintaining stability for a particular constraint condition and an adaptive control algorithm be developed to adjust the program force error gains to maintain β in a stable region. This requires a means of evaluating the constraint stiffness in real time during manipulator motion. The existing control system must be modified to provide the ability to measure and utilize the actual forces acting at the

97

end effector, not the force error between the end effector and joystick. With knowledge of the actual forces, the stiffness of the environmental constraint can be continuously calculated in each direction within the control algorithm. This stiffness can then be correlated to a satisfactory control gain for maintaining stability, similar to what is performed in the compliance controller developed in [Ref. 9].

2. Computational Delays

Computational delays have a very destabilizing effect on a control system. The delays in this system drives an inherently stable response of a system with no delays to unstable behavior in some situations. Building the force override controller on top of the existing PUMA control system greatly limits efforts to minimize the delay time. The performance of the system is most limited by the structure of the continuous path motion control algorithm of the manipulator. Being a position-based control system, the PUMA's performance is worse when no effort is made to ensure continuous path motion is maintained while the manipulator is moving in free space. The additional time required for acceleration and deceleration of the manipulator slows down the overall performance and results in jerky motion. Similar operating constraints will exist for other positioned-based control systems.

C. COMPARISON BETWEEN SIMULATION AND ACTUAL SYSTEM

In general, the response of the simulation results are more rapid than the response of the actual system for a step commanded force. Compare the simulation responses shown in Figure 4.24 with those of Figures 4.9, 4.10, and 4.11 as well as those of Figure 4.25 with Figures 4.12, 4.13, and 4.14 and the differences in the rate of response between the simulation and the actual system are readily apparent. Comparing Figures 4.26 and 4.27 with Figures 4.19 and 4.20, this is also the case for the sinusoidal responses, but not nearly as noticeable. Several factors contribute to the differences in response. The end effector stiffness used for the simulation are only estimates of the actual end effector stiffness. An actual end effector stiffness much lower than the estimated stiffness used in the control program and the simulation would explain much of the difference between the actual results and the simulation. Additionally, calibration errors and the varying and the lack of an exact speed of motion, as discussed above, are not simulated and contribute to the discrepancies between the simulation and actual results. The discrepancies can not be fully explained without better knowledge of the actual system parameters.

The simulation results depict the performance of the actual system effectively. For the step commanded force the response of the system as it begins unstable motion is simulated well, as shown by Figures 4.11 and 4.24. The

simulation response of Figure 4.26 does not depict the unstable motion of the actual system in Figure 4.18 for the sinusoidal force input nearly as well. The simulation algorithm successfully models the general performance of the system but requires improvements to more accurately predict the actual system response. The algorithm can be extended to provide a six DOF model of the manipulator. In this model each link would be considered as a dynamic system instead of lumping them all into one system as has been done for this simple model. This would require much more knowledge of the manipulator's characteristics and many more computations to simulate the manipulator's motion. The interaction forces between the links and with environmental constraints can be modelled to varying degrees of accuracy. If speeds and accelerations are expected to be very low Equation 2.9 can be used to determine the reaction forces between the links and at the end effector. If high speeds or high accelerations are expected more complex equations of motion must be developed to model the manipulator's dynamics. Methods for calculating and modelling the dynamics are discussed in [Ref. 28] and [Ref. 29].

VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

- Three axis unilateral force override rate control has been successfully implemented using the PUMA 560 manipulator.

- Stability of the force override rate control system is dependent on the stiffness of the manipulator, the end effector, and the environmental constraints.

- Steady state and transient performance is limited due to a loss of stability with high stiffness and/or high control gains.

- The system stability is limited due to the use of constant feedback gains in the joint servo controllers.

- The response of the PUMA 560 force override rate control system performance has been successfully modelled for a single DOF using a simple computer model.

B. RECOMMENDATIONS

- Develop and construct a force torque sensor and associated hardware utilizing force sensing resistors (FSR) to replace the force sensor currently used for the joystick and end effector.

- Extend the control system to control not only forces but also torques, ultimately leading to six DOF force-torque override of linear and angular rate control of a manipulator.

- Develop a force-torque display to assist an operator in controlling and monitoring system conditions.

- Study the feasibility of using an adaptive control algorithm which uses real time force-torque data to adjust system gains in order to maintain system stability.

APPENDIX A

```
 1          TYPE "This program is written to implement force control of the"
 2          TYPE "PUMA with the proper hardware installed."
 3          TYPE " "
 4          TYPE " "
 5          TYPE "Do you want to use integral control in the position"
 6          TYPE "control of the joint servos (Warning-excessive control"
 7          TYPE "may develop if large forces or high stiffness is present)"
 8      10  PROMPT "Enter '1' for'YES' and '2' for 'NO'",answer
 9          IF answer == 1 THEN
10            dumy = 1
11          ELSE
12            IF answer == 2 THEN
13              INTOFF ALWAYS
14            ELSE
15              IF answer <> 1 THEN
16                GOTO 10
17              END
18            END
19          END
20          TYPE " "
21          TYPE " "
22          NONULL ALWAYS
23          adfactor = (5/255)*(12/2.5)
24          xamp.gain = 1.80
25          yamp.gain = 2.75
26          zamp.gain = 0.6745
27          TYPE "Enter the factor 'beta' that is divided by the estimated"
28          TYPE "environmental and end effector gain (Ke) to give the force"
29          TYPE "error gain (kf) in the x, y, and z directions."
30          TYPE " "
31          PROMPT "? ", betax, betay, betaz
32          xforce.gain = betax/1.205
33          yforce.gain = betay/0.525
34          zforce.gain = betaz/104.8
35          TYPE "Enter the desired values for the force integral control"
36          TYPE "gains for the x, y, and z directions respectively if"
37          TYPE "integral control is desired. Enter zeros (0) if no"
38          PROMPT "integral control is desired ? ",xint.gain, yint.gain, zint.gain
39          TYPE " "
40          TYPE " "
41          xforcecon = adfactor/xamp.gain
42          yforcecon = adfactor/yamp.gain
43          zforcecon = adfactor/zamp.gain
```

```
44          movetime = .4E-1
45          maxspeed = 462
46          intxferr = 0
47          intyferr = 0
48          intzferr = 0
49          TYPE "The strain gage circuit is being calibrated  now"
50          TYPE "Hit 'RETURN' when the joystick and endeffector  are"
51          PROMPT "clear of any external forces", dumy
52          SIGNAL 1, 2, 3, 4, -5, -6, -7, -8
53          FOR i = 1 TO 5
54             delay = 1
55          END
56          xcal = BITS(1001, 8)
57          SIGNAL -1, 2, 3, 4, -5, -6, -7, -8
58          FOR i = 1 TO 5
59             delay = 1
60          END
61          ycal = BITS(1001, 8)
62          SIGNAL 1, -2, 3, 4, -5, -6, -7, -8
63          FOR i = 1 TO5
64             delay = 1
65          END
66          zcal = BITS(1001, 8)
67          TYPE " "
68          TYPE "Calibration  is now complete; Proceeding."
69          TYPE " "
70          TYPE "You my modify the system to establish  desired force"
71          PROMPT "input. Press 'RETURN' when ready.",dumy
72     50   SIGNAL 1, 2, 3, 4, -5, -6, -7, -8
73          FOR i = 1 TO 5
74             delay = 1
75          END
76          xin = BITS(1001, 8)
77          SIGNAL -1, 2, 3, 4, -5, -6, -7, -8
78          FOR i = 1 TO 5
79             delay = 1
80          END
81          yin = BITS(1001, 8)
82          SIGNAL 1, -2, 3, 4, -5, -6, -7, -8
83          FOR i = 1 TO 5
84             delay = 1
85          END
86          zin = BITS(1001, 8)
87          xerr = xcal-xin
88          yerr = ycal-yin
89          zerr = zcal-zin
90          xferr = xerr*xforcecon
91          yferr = yerr*forcecon
92          zferr = -(zerr+14.3/12)*zforcecon
93          xmove = xforce.gain*(xferr+intxferr*xint.gain)
94          ymove = yforce.gain*(yferr+intyferr*yint.gain)
```

```
95      zmove  = zforce.gain*(zferr+intzferr*zint.gain)
96      distance  = SQRT(SQR(xmove)+SQR(ymove)+SQR(zmove))
97      velocity  = distance/movetime
98      SPEED  = 100/SPEED(1)*100/maxspeed*velocity
99      SET  delta  = TRANS(xmove,  ymove, zmove, 90,-90,0)
100     HERE initial
101     MOVE initial:delta
102     intxferr  = intzferr+xferr*movetime
103     intyferr  = intyferr+yferr*movetime
104     intzferr  = intzferr+zferr*movetime
105     GOTO 50
```

APPENDIX B

'''
'   '
'   1 DOF Force Control Simulation   '
'   '
'   of PUMA 560 Robot   '
'   '
'   Larry P.Ondrey   '
'   '
'   Revised: 1/25/93   '
'   '
'''
' List of Variables: '
'   '
' inertia : Estimate of the inertia of the PUMA   '
'   manipulator.(kg)   '
' ki.pos : Estimate of the position error integral gain in   '
'   the servo control loop of the PUMA.   '
' kp : Estimate of the position error gain in the servo   '
'   control loop of the PUMA (N/m).   '
' damp : Estimate of the damping term in the PUMA's   '
'   position control system; assumed to provide   '
'   critical damping.   '
' kend : Estimate of the end effector's stiffness (N/m).   '
' kf :   Force error gain used in the PUMA force   '
'   control algorithm (m/N).   '
' ki.force : Force error integral gain used in the PUMA   '
'   force control algorithm (m/N-sec).   '
' accelmax : Estimated maximum acceleration of the PUMA   '
'   endeffector in cartesian space (m/sec^2).   '
' maxvel : Estimated maximum veloctiy of the PUMA end-   '
'   effector in cartesian space (m/sec).   '
' rpos : Actual position of robot tool mount (m).   '
' veldes : Commanded velocity from the control algorithm   '
'   based on the force error (m/sec).   '
' epos : Position of rigid environment (m).   '
' fdes :   Commanded force from joystick (N).   '
' accel : Actual accel. of the PUMA's endeffector (m/sec^2).   '
' velocity : Actual vel. of PUMA endeffector (m/sec).   '
' fact : Actual force being applied by PUMA's endeffector   '
'   on the environment (N).   '
' time : Elapsed time of system conditions from the the   '
'   conditions at time=0 (sec).   '
' tstep : Time increment used in simulation for integration   '
'   steps. Based on loop time of the PUMA's individual   '
'   servo control loops - .875 msec.   '

```
' k : Sampling counter used to limit the number of          '
'     data points stored in the data file - Based on        '
'     the actual sampling rate of the data aquisition system. '
' ferror : The force error between the desired force and the '
'         actual force at a particular time.(N)             '
' nextmove : The next commanded move based on the current    '
'           force error. (m)                                '
' desvel.next : The next commanded velocity based on the     '
'             current force error.(m/sec)                   '
' despos.next : The next position the control system is      '
'             commanded to go to. (m)                       '
' reqtime : The estimate of the time that the current move is '
'         going to take based on desired speed and distance.'
'         (sec)                                             '
' cycles : The number of cycles the PUMA's main controller   '
'         will use to command the joint servo controllers to '
'         perform the current move.                         '
' intposerr : The integral of the position error; used in the'
'           joint servo controllers to prevent S.S errors   '
'           in the position error (m-sec)                   '
' despos : The current commanded position that the PUMA      '
'         main controller is trying to achieve.(m)          '
' midpos : The incremental position the main controller      '
'         sends to the individual joint controller every    '
'         28 msec.                                          '
' desvelocity : Estimate of the current commanded            '
'             velocity that the PUMA main controller is     '
'             trying to achieve.(m/sec)                     '
' i : Counter for the number of cycles the mian controller   '
'     has gone through with the joint servos.               '
' j : Counter for the servo controller loop.                '
' poserr : The current position error used in the servo      '
'         controller loop to control the commanded          '
'         force.(m)                                         '
' fcomm : The current commanded force that the servo         '
'         controller sends to the servo motor.(N)           '
' ferrorint : The integral of the force error.(N-m)          '
'                                                          '

'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
'

CLS
PRINT
PRINT
PRINT "This program simulates the dynamics of the force control"
PRINT
PRINT "system of the PUMA robot."
PRINT
OPEN "c:\thesis\pumasim.dat"  FOR OUTPUT AS #1

'Initialize system "constants":
```

```
   inertia  = 5.38
   kp = 85000
   damp = 1! * 2 * SQR(kp * inertia)
   accelmax  = 4.12
   maxvel  = .462
'
PRINT
PRINT "Enter the force gain (kf), Force integral gain (ki.force),"
PRINT "position  integral  gain  (ki.pos), and end effector "
INPUT "stiffness  (N/m)  "; kf, ki.force, ki.pos, kend
PRINT
PRINT
kf = 1 / kf
'
'Intialize  system  varibles:
'
   rpos = 0
   midpos  = 0
   veldes  = 0
   epos = .001
   fdes = 1.9 + 1.6 * SIN(7.85 * 1.5)
   accel = 0
   velocity  = 0
   fact = 0
   time = 0
   tstep = .875 / 1000
   k = 0
   ferrorint  = 0
'
' Store  initial  conditions  in the  data file.
'
   PRINT #1, USING "##.####,"; time; fdes; fact; rpos
'
' Establish  initial  control  variable  values.
'
   ferror = fdes - fact
   nextmove  = kf * ferror
   desvel.next  = nextmove  / .14
   IF desvel.next  > maxvel  THEN
      desvel.next  = SGN(desvel.next)  * maxvel
   END IF
   despos.next  = rpos + nextmove
'
' Begin  simulation  loop.
'
WHILE time < 6.5
   despos  = despos.next
   desvel  = desvel.next
   reqtime  = ABS(nextmove  / desvel)
   cycles = INT(reqtime  / .028)
   IF cycles < 2 THEN
```

```
            cycles = 2
         END IF
      FOR i = 1 TO cycles
         intposerr  = 0
         midpos = rpos + desvel * .028
'

         FOR j = 1 TO 32
            k = k + 1
            time = time + tstep
            fdes = 1.9 + 1.6 * SIN(7.8 * (time + 1.5))
            poserr = midpos - rpos
            fcomm = kp * poserr + ki.pos * intposerr
            accel = 1 / inertia * (fcomm - fact) - velocity * damp / inertia
            IF ABS(accel) > accelmax  THEN
               accel = SGN(accel) * accelmax
            END IF
'

' Integration  steps
'

            intposerr = intposerr + poserr * tstep
            rpos = rpos + velocity * tstep
            velocity = velocity + accel * tstep
            IF velocity > desvel THEN
               velocity = desvel
            END IF
            IF rpos < epos THEN
               fact = 0
            ELSE
               fact = kend * (rpos - epos)
            END IF
            IF k = 13 THEN
               k = 0
               PRINT #1, USING "##.####,"; time; fdes; fact; rpos
            END IF

         NEXT j
         IF i = 1 THEN
            ferror = fdes - fact
            ferrorint = ferrorint + ferror * .028 * cycles
            nextmove = kf * ferror + ki.force * ferrorint
            desvel.next = nextmove / .14
            IF desvel.next > maxvel THEN
               desvel.next = SGN(desvel.next) * maxvel
            END IF
            despos.next = rpos + nextmove
         END IF
'

   NEXT i
'

WEND
'
```

```
CLOSE #1
BEEP
PRINT "Simulation  is complete. Data  is stored in"
PRINT
PRINT "file PUMASIM.dat."
```

LIST OF REFERENCES

1. Syvertsen, J.M., *Force Override Rate Controller for Remote Actuation*, Masters Thesis, Naval Postgraduate School, September, 1992.

2. Fu, K.S., Gonzalez, R.C., and Lee, C.S.G., *Robotics: Control, Sensing, Vision, and Intelligence*, pp.210-220, McGraw-Hill, New York, 1987.

3. Craig, J.J, *Introduction to Robotics: Mechanics and Control, Second Edition*, pp. 196-210, Addison-Wesley,Reading, MA, 1989.

4. Fu, K.S., Gonzalez, R.C., and Lee, C.S.G., *Robotics: Control, Sensing, Vision, and Intelligence*, pp. 244-263, McGraw-Hill, New York, 1987.

5. Craig, J.J., *Introduction to Robotics: Mechanics and Control, Second Edition*, pp.353-358, Addison-Wesley, Reading, MA, 1989.

6. Fu, K.S., Gonzalez, R.C., and Lee, C.S.G., *Robotics: Control, Sensing, Vision, and Intelligence*, pp. 232-244, McGraw-Hill, New York, 1987.

7. Fu, K.S., Gonzalez, R.C., and Lee, C.S.G., *Robotics: Control, Sensing, Vision, and Intelligence*, p. 242, McGraw-Hill, New York, 1987.

8. Whitney, D.E., "Historical Perspective and State of the Art in Robot Force Control," *Proceedings of the IEEE Conference on Robotics and Automation*, (1985), pp. 262-268.

9. Ishikawa, H., Sawada, C., Kawase, K., and Takata, M., "Stable Compliance Control and Its Implementation for a 6 D.O.F. Manipulator," *Proceedings of the IEEE Conference on Robotics and Automation*, (1989), pp. 98-102

10. Ishikawa, H., Sawada, C., Kawase, K., and Takata, M., "Stable Compliance Control and Its Implementation for a 6 D.O.F. Manipulator," *Proceedings of the IEEE Conference on Robotics and Automation*, (1989), p. 102.

11. Eppinger, S.D., and Seering, W.P., "Understanding Bandwidth Limitations in Robot Force Control," *Proceedings of the IEEE Conference on Robotics and Automation*, (1987), pp.904-909.

12. Syvertsen, J. M., *Force Override Rate Controller for Remote Actuation*, pp. 5-12, Master's Thesis, Naval Postgraduate School, September 1992.

13. Syvertsen, J. M., *Force Override Rate Controller for Remote Actuation*, pg. 6, Master's Thesis, Naval Postgraduate School, September 1992.

14. Merritt, H.E., *Hydraulic Control Systems*, pp.145-152, John Wiley and Sons, 1967.

15. Driels, M.R., *Force Overide Rate Controller for Remote Actuation: Final Report FY92*, pg. 23, Unpublished report, Naval Postgraduate School, 1992.

16. Driels, M.R., *Force Overide Rate Controller for Remote Actuation: Final Report FY92*, pp. 8-11, Unpublished report, Naval Postgraduate School, 1992.

17. National Semiconductor Corporation, *Linear Databook*, pp. 8-71 - 8-81, National Semiconductor Corporation, 1982.

18. Driels, Morris R., *Force Overide Rate Controller For Remote Actuation: Final Report FY92*, pg. 8, Naval Postgraduate School, 1992.

19. Driels, Morris R., *Force Overide Rate Controller For Remote Actuation: Final Report FY92*, pg. 24, Unpublished report, Naval Postgraduate School, 1992.

20. Driels, Morris R., *Force Overide Rate Controller For Remote Actuation: Final Report FY92*, pg. 25, Unpublished report, Naval Postgraduate School, 1992.

21. Fu, K.S., Gonzalez, R.C., Lee, C.S., *Robotics: Control, Sensing, Vision, and Intelligence*, pp. 203-204, McGraw-Hill, New York, 1987.

22. Unimation Incorporated, *Unimate PUMA Mark II Robot 500 Series Equipment Manual for VAL II and VAL PLUS Operating Systems*, pp. 1-1 - 1-6, Unimation Incorporated, Danbury, CT, August 1985.

23. Unimation Incorporated, *Unimate PUMA Mark II Robot 500 Series Equipment Manual for VAL II and VAL PLUS Operating*

Systems, pg. 1-9, Unimation Incorporated, Danbury, CT, 1985.

24. Unimation Incorporated, *Unimate Industrial Robot Programming Manual: User's Guide to VAL II Version 2.0 Part 1 - Control from the System Terminal*, pg. 7-13, Unimation Incorporated, Danbury, CT, February, 1986.

25. Unimation Incorporated, *Unimate Industrial Robot Programming Manual: User's Guide to VAL II Version 2.0 Part 1 - Control from the System Terminal*, pg. 7-14, Unimation Incorporated, Danbury, CT, february, 1986.

26. Unimation Incorporated, *Unimate PUMA mark II Robot 500 Series Equipment Manual for VAL II and VAL PLUS Operating Systems*, pp. 1-29 - 1-32, Unimation Incorporated, Danbury, CT, 1985.

27. Unimation Incorporated, *Unimate Industrial Robot Programming Manual: User's Guide to Val II Version 2.0 Part 1 - Control from the System Terminal*, pg. 8-32, Unimation Incorporated, Danbury, CT, February, 1986.

28. Fu, K.S., Gonzalez, R.C., Lee, C.S., *Robotics: Control, Sensing, Vision, and Intelligence*, pp.82-142,McGraw-Hill, New York, 1987.

29. Craig, J.J, *Introduction to Robotics: Mechanics and Control, Second Edition*, pp. 187-219, Addison-Wesley, Reading, MA, 1989.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 052 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Mr. L. Monford NASA Johnson Space Center Mail Stop ER Houston, TX 77058	1
4.	Naval Engineering Curricular Office Code 34 Department of Mechanical Engineering Naval Postgraduate School Monterey, CA 93943-5002	1
5.	Department Chairman, Code ME Department of Mechanical Engineering Naval Postgraduate School Monterey, CA 93943-5000	1
6.	Professor Morris Driels, Code ME/Dr Department of Mechanical Engineering Naval Postgraduate School Monterey, CA 93943-5000	3
7.	LT James M. Syvertsen 20 Leonard Street Wading River, NY 11792	1
8.	LT Larry P. Ondrey 608 Ryder Cup Lane Virginia Beach, VA 23462	3